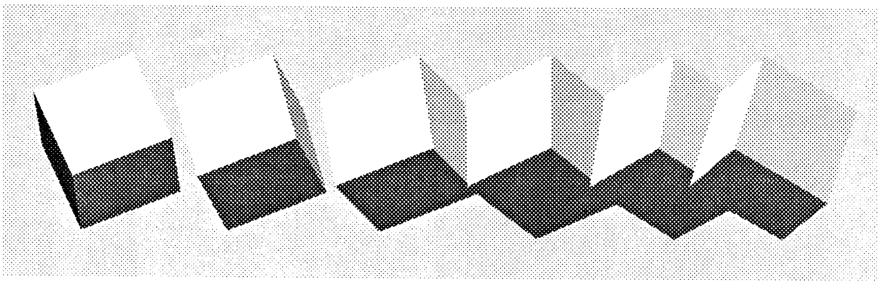


Chapter 3

The dimension weighting models

3.1 Introduction

In the previous chapter we discussed how to match n configurations when the only admissible transformations are those that leave relative distances intact. We also discussed that the transformations satisfying this restriction are translations, orthonormal transformations, and uniform rescalings of the n configurations. In this chapter we investigate the matching of n configurations using a less restrictive set of rules. Specifically, the condition that the configurations may only be rescaled uniformly is relaxed to the effect that *each of the m dimensions* of each configuration is allowed to be stretched or shrunk *differently*. The latter transformation is called dimension weighting, and is illustrated for one two-dimensional configuration in Figure 3.1, where the stimulus points are numbered from one to five before transformation and from one prime to five prime after transformation. As the figure shows, relative distances are no longer preserved in dimension weighting.

This chapter deals with the matching of n configurations when the admissible transformations are translations, orthonormal transformations, and dimension weighting. Lingoes and Borg (1978) discussed how these three transformations quite naturally result in two different so-called dimension weighting models. As we already noted in chapter 1, the way in which Lingoes and Borg developed these two models has an important drawback: the estimation of some of the unknown transformation parameters is based on results derived for two-dimensional configurations only. In the following sections we will not only generalize the dimension weighting models of Lingoes and Borg to m dimensions, but also to the case of missing data. Moreover, a

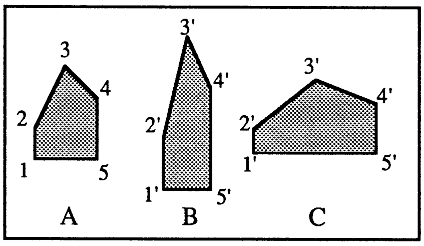


Figure 3.1 Illustration of dimension weighting. A: before weighting, B and C: after differential weighting of the two dimensions of A.

new approach to the fitting of these models will be proposed. While Lingoes and Borg chose to hold on to the centroid configuration **Z** obtained in the GPA model in the development of their dimension weighting models, we propose to determine *new* optimal centroid configurations. This results in a better fit, and yields models that can, at least in principle, be fitted independently from any results obtained in GPA. In the following section we will first explain how Lingoes and Borg set up their models, and then discuss our alternative approach.

3.2 Geometry and algebra

Given that the admissible transformations in the matching of n configurations are translations, orthonormal transformations, and dimension weighting, Lingoes and Borg (1978) proposed to apply these transformations in the following way. Given n configurations $\mathbf{X}_j$ (j = 1, ..., n) of order (p x m), let **Z** of order (p x m) be the matrix of centroids obtained after a generalized Procrustes analysis of these n configurations. Also, let $\mathbf{g}_j$ and $\mathbf{h}_j$ be unknown translation vectors of order (m x 1), $\mathbf{Q}_j$ be an unknown orthonormal matrix of order (m x m), $\mathbf{W}_j$ be an unknown (m x m) diagonal matrix containing dimension weights, and $\mathbf{E}_j$ be a (p x m) matrix of residuals. Using these definitions, Lingoes and Borg proposed the following model:

$$(\mathbf{X}_j - \mathbf{1}\mathbf{g}_j')\mathbf{Q}_j = (\mathbf{Z} - \mathbf{1}\mathbf{h}_j')\mathbf{W}_j + \mathbf{E}_j, \qquad \text{for } j = 1, 2, ..., n, \tag{3.1}$$

where the unknown $\mathbf{g}_j$'s, $\mathbf{h}_j$'s, $\mathbf{Q}_j$'s, and $\mathbf{W}_j$'s are to be estimated by minimizing the sum of squared residuals, that is, by minimizing $\Sigma \operatorname{tr} \mathbf{E}_j'\mathbf{E}_j$.

Lingoes and Borg discussed that, to obtain optimal results, it is necessary to include still another (admissible) transformation in (3.1). They showed that a rotation of the dimensions of **Z** is essential to get the best out of differential weighting of these same dimensions. A simple geometrical example may illustrate the necessity of an optimal orientation of **Z**. Suppose that the coordinates of the stimuli in **Z** are the vertices of the square shown in Figure 3.2. Also suppose that the four stimuli in one of the individual configurations $\mathbf{X}_j$ form the vertices of the rhombus shown in the same figure. By differential weighting of the two dimensions of the centroid configuration **Z** all kinds of rectangles can be obtained, but never the rhombus representing $\mathbf{X}_j$. If only weights are allowed, therefore, the match between $\mathbf{X}_j$ and **Z**

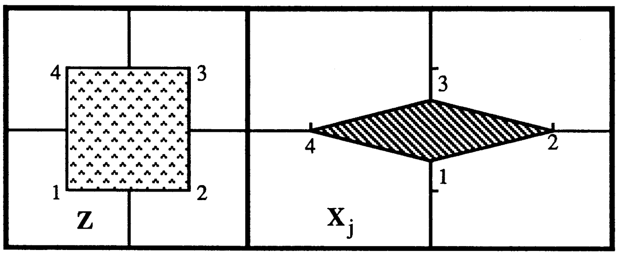


Figure 3.2 Illustration of necessity of rotation of **Z** with respect to dimension weighting.

will never be perfect. But it is immediately apparent how a rotation of **Z** before weighting its dimensions can effectuate a perfect match between $\mathbf{X}_j$ and **Z**. Rotating **Z** counterclockwise through an angle of 45°, and then weighting the first dimension of **Z** with a factor $\sqrt{2}$ and the second dimension with a factor $(1/4)\sqrt{2}$ exactly gives the rhombus representing $\mathbf{X}_j$ in Figure 3.2. In matrix notation this can be expressed as follows:

$$\mathbf{ZSW} = \begin{bmatrix} -1 & -1 \\ 1 & -1 \\ 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} \cos 45° & \sin 45° \\ -\sin 45° & \cos 45° \end{bmatrix} \begin{bmatrix} \sqrt{2} & 0 \\ 0 & (1/4)\sqrt{2} \end{bmatrix}$$

$$= \begin{bmatrix} -1 & -1 \\ 1 & -1 \\ 1 & 1 \\ -1 & 1 \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ -1/\sqrt{2} & 1/\sqrt{2} \end{bmatrix} \begin{bmatrix} \sqrt{2} & 0 \\ 0 & (1/4)\sqrt{2} \end{bmatrix} = \begin{bmatrix} 0.0 & -0.5 \\ 2.0 & 0.0 \\ 0.0 & 0.5 \\ -2.0 & 0.0 \end{bmatrix} = \mathbf{X}_j,$$

where **S** is the optimal rotation matrix, and **W** contains the optimal dimension weights to achieve a perfect match between **Z** and $\mathbf{X}_j$. These transformations of **Z** are also illustrated geometrically in Figure 3.3.

Lingoes and Borg next showed that such an orthonormal transformation of **Z**, needed to obtain an optimal match in model (3.1), can be performed in two ways, and that from these two options two different dimension weighting models emerge.

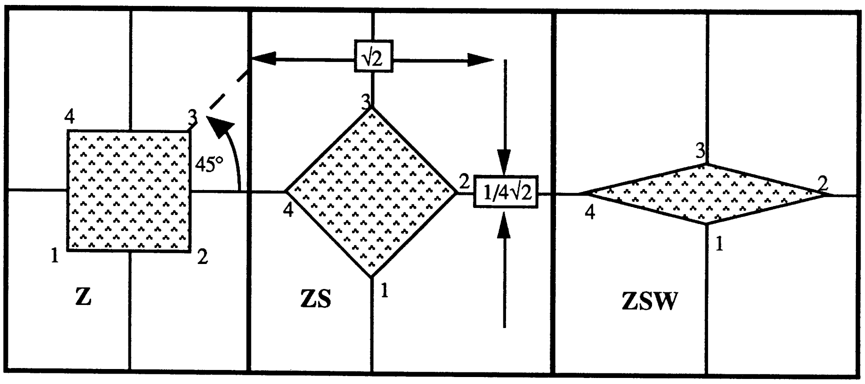


Figure 3.3 Geometry of optimal rotation of **Z** in Figure 3.2 with respect to dimension weighting.

The first way to rotate **Z** is to estimate only one orthonormal transformation of **Z** such that the match is optimized over all n individual configurations simultaneously. The geometry of this model has a rather straightforward psychological interpretation: people (represented by the individual configurations) agree on the underlying dimensions that structure the surrounding world (represented by the optimally rotated centroid or group configuration), but differ in the importance they attach to the dimensions on which the stimuli under investigation are ordered. The algebra corresponding to this model is as follows:

$$(\mathbf{X}_j - \mathbf{1g}_j')\mathbf{Q}_j = (\mathbf{Z} - \mathbf{1h}_j')\mathbf{SW}_j + \mathbf{E}_j \quad \text{for } j = 1, ..., n, \tag{3.2}$$

where **S** is an unknown orthonormal matrix of order (m x m). The matrices $\mathbf{X}_j$, **Z**, $\mathbf{W}_j$, and $\mathbf{E}_j$, and the vectors $\mathbf{g}_j$ and $\mathbf{h}_j$ have been defined earlier in this section.

This model is related to the INDSCAL model proposed by Carroll and Chang (1970). Both model (3.2) and the INDSCAL model are characterized by a group configuration whose dimensions are idiosyncratically being weighted. Therefore, these models have a common psychological interpretation. Here the resemblance stops, however, since Lingoes and Borg's model was designed to match n *configurations* of order (p x m), while Carroll and Chang use the INDSCAL model to analyse n *(dis)similarity matrices* of order (p x p).

The second way to rotate **Z** is to estimate a separate optimal orthonormal transformation for each of the n individual configurations. The psychological implication of this second dimension weighting model is that individuals not only differ in the importance they attach to the dimensions in the group space, but also differ in which dimensions they choose to weight in this same group configuration. The matrix notation for this model is very similar to (3.2):

$$(\mathbf{X}_j - \mathbf{1g}_j')\mathbf{Q}_j = (\mathbf{Z} - \mathbf{1h}_j')\mathbf{S}_j\mathbf{W}_j + \mathbf{E}_j \quad \text{for } j = 1, ..., n, \tag{3.3}$$

the only difference being that the fixed orientation matrix S has now been replaced by the idiosyncratic orientation matrix $\mathbf{S}_j$. With respect to interpretation, this model is related to the IDIOSCAL model presented by Carroll and Chang in 1972. Again, however, model (3.3) was set up for the analysis of configurations, while Carroll and Chang's IDIOSCAL model was designed for the analysis of (dis)similarity matrices.

What is striking about the way Lingoes and Borg set up these two models is that they chose to 'borrow' the centroid configuration **Z** from GPA. The fact that **Z** is optimal in GPA, however, does not guarantee that it is optimal in models (3.2) and (3.3). This means that by sticking to **Z** a source of error is introduced obscuring the real fit of these models. This is the reason why we propose to replace models (3.2) and (3.3) of Lingoes and Borg by two models that always find the perfect solution if it exists, and eliminate error due to compromising factors resulting from GPA. Instead of using the matrix of centroids **Z** obtained in the GPA model, a *new* optimal centroid configuration **Y** is estimated.

Taking into account that information may be missing about some stimuli in some configurations, and letting $\tilde{\mathbf{X}}_j$ denote the optimally translated, rotated and rescaled configuration $\mathbf{X}_j$ after GPA rotated to the principal components of **Z** (i.e., $\tilde{\mathbf{X}}_j = s_j(\mathbf{X}_j - \mathbf{1u}_j')\mathbf{R}_j\mathbf{K}$), we propose the following alternative for model (3.2):

$$\mathbf{M}_j(\tilde{\mathbf{X}}_j - \mathbf{1g}_j')\mathbf{Q}_j = \mathbf{M}_j(\mathbf{Y} - \mathbf{1h}_j')\mathbf{W}_j + \mathbf{M}_j\mathbf{E}_j \quad \text{for } j = 1, ..., n. \tag{3.4}$$

In (3.4), **Y** is assumed to be an unknown centroid configuration **Y** of order (p x m), and $\mathbf{M}_j$ is a given diagonal matrix of order (p x p) with ones on the diagonal if the corresponding rows in $\mathbf{X}_j$ are not missing, and zeroes elsewhere. The reason that we use the optimally transformed configurations $\tilde{\mathbf{X}}_j$ instead of the raw configurations $\mathbf{X}_j$ in model (3.4) is a practical one. We simply expect that the algorithm to be developed in section 3.3 will be more efficient when applied to the $\tilde{\mathbf{X}}_j$'s than to the raw $\mathbf{X}_j$'s, because the former configurations are already optimal with respect to distance preserving transformations. It may be noted that the orthonormal matrix **S** is lacking in model (3.4). This latter matrix no longer needs to be estimated since the free matrix of centroids **Y** will automatically be in an optimal orientation with respect to dimension weighting. In full we call this alternative model the *dimension weighting model with free optimal centroid configuration*, and will refer to it as the DIMFREE model. The DIMFREE model is completely developed in section 3.3 of this chapter.

Our alternative for model (3.3) can be written as follows:

$$\mathbf{M}_j(\tilde{\mathbf{X}}_j - \mathbf{1g}_j')\mathbf{Q}_j = \mathbf{M}_j(\mathbf{Y} - \mathbf{1h}_j')\mathbf{S}_j\mathbf{W}_j + \mathbf{M}_j\mathbf{E}_j \quad \text{for } j = 1, ..., n, \tag{3.5}$$

where **Y** is, again, assumed to be an unknown (p x m) centroid configuration. In full we call this model the *dimension weighting model with idiosyncratic optimal orientations of a free optimal centroid configuration.* Henceforth we will refer to this model as the DIMIDIO model. The DIMIDIO model is completely developed in section 3.4 of this chapter.

A characteristic feature of all dimension weighting models discussed so far is that the weighting of dimensions is performed on the centroid or group configuration (whether it be **Y** or **Z**), and not on the individual configurations. In the GPA model, on the other hand, it is typically the individual configurations that are optimally being rescaled. The choice of Lingoes and Borg to weight the centroid configuration instead of the individual configurations was probably based on traditional considerations: in the dimension weighting models INDSCAL and IDIOSCAL of Carroll and Chang (1970, 1972), it is also the dimensions of the group configuration that are being weighted. In the latter models this manner of weighting is born out of necessity, since it does not make sense to weight the dimensions of a (dis)similarity matrix. When analyzing configurations, however, there is, in principle, nothing against using a model like, for instance,

$$\mathbf{M}_j(\tilde{\mathbf{X}}_j - \mathbf{1g}_j')\mathbf{Q}_j\mathbf{W}_j = \mathbf{M}_j\mathbf{Y} + \mathbf{M}_j\mathbf{E}_j, \quad \text{for } j = 1, 2, ..., n, \tag{3.6}$$

where the dimensions of the individual configurations are weighted instead of those of **Y**. In that case we could again apply Theorem 2 (see section 2.2), and show that the minimization of the sum of squared distances between corresponding stimulus points is equivalent to the minimization of the sum of squared distances between corresponding points and their centroid. But we will not follow this line of reasoning any further, and use the 'traditional' weighting of the dimensions of the centroid configuration.

The remainder of this chapter is organized as follows. In sections 3.3 and 3.4 the dimension weighting models (3.4) and (3.5) are developed and algorithms for the estimation of the corresponding unknown transformation parameters are presented. In section 3.5 measures of fit and an analysis of variation are presented for both models, and in section 3.6 we discuss the results of the analysis of constructed data sets according to the DIMFREE and DIMIDIO model.

3.3 The DIMFREE model

3.3.1 Introduction

To determine the unknown parameters in the DIMFREE model (3.4) the following least squares loss function

$$f(\mathbf{G},\mathbf{H},\mathbf{Q},\mathbf{W},\mathbf{Y}) = \sum_{j=1}^{n} \text{tr } \mathbf{E}_j'\mathbf{M}_j\mathbf{E}_j$$

$$= \sum_{j=1}^{n} \text{tr } [(\tilde{\mathbf{X}}_j - \mathbf{1g}_j')\mathbf{Q}_j - (\mathbf{Y} - \mathbf{1h}_j')\mathbf{W}_j]'\mathbf{M}_j[(\tilde{\mathbf{X}}_j - \mathbf{1g}_j')\mathbf{Q}_j - (\mathbf{Y} - \mathbf{1h}_j')\mathbf{W}_j] \quad (3.7)$$

is defined. The unknown translation vectors $\mathbf{g}_j$ and $\mathbf{h}_j$ are collected in the (m x n) matrices $\mathbf{G}$ and $\mathbf{H}$, respectively, the unknown orthonormal matrices $\mathbf{Q}_j$ in the (nm x m) supermatrix $\mathbf{Q}$, and the unknown dimension weight matrices $\mathbf{W}_j$ in the (nm x m) supermatrix $\mathbf{W}$. In section 3.3.2 we first of all determine the translation vectors $\mathbf{g}_j$ optimizing (3.7), since this makes it possible to simultaneously eliminate $\mathbf{G}$ and $\mathbf{H}$ from (3.7). In sections 3.3.3, 3.3.4 and 3.3.5 the simplified loss function for the DIMFREE model is minimized with respect to the orthonormal matrices $\mathbf{Q}_j$, dimension weights $\mathbf{W}_j$ and new centroid configuration $\mathbf{Y}$, respectively. In section 3.3.6 we briefly go into the possibility of setting up a 'direct approach' for the DIMFREE model by eliminating the unknown matrix $\mathbf{Y}$ from the DIMFREE loss function, and discuss the consequences this has for the optimization of the resulting loss function with respect to the only unknowns left: the orthonormal matrices $\mathbf{Q}_j$, and the dimension weights $\mathbf{W}_j$. In section 3.3.7 we show that the solution for the DIMFREE model is only unique up to a simultaneous weighting of the optimal $\mathbf{Y}$ and $\mathbf{W}_j$'s in (3.7), and discuss a procedure which guarantees a uniquely weighted solution. Finally, in section 3.3.7 an algorithm is presented for the fitting of the DIMFREE model to n configurations.

3.3.2 Translations

In this section we minimize (3.7) with respect to the unrestricted translation vectors **G**. Defining $\mathbf{A}_j = \tilde{\mathbf{X}}_j\mathbf{Q}_j - (\mathbf{Y} - \mathbf{1h}_j')\mathbf{W}_j$, and considering only one particular $\mathbf{g}_j$, we may rewrite the loss function as

$$f(\mathbf{g}_j) = \mathbf{1'M}_j\mathbf{1g}_j'\mathbf{g}_j - 2\,\mathbf{g}_j'\mathbf{Q}_j\mathbf{A}_j'\mathbf{M}_j\mathbf{1} + d_j$$

$$= c_j^2\,\mathbf{g}_j'\mathbf{g}_j - 2\,c_j\mathbf{g}_j'\mathbf{b}_j + d_j, \tag{3.8}$$

where d_j is a term independent of $\mathbf{g}_j$, $\mathbf{b}_j \equiv \mathbf{Q}_j\mathbf{A}_j'\mathbf{M}_j\mathbf{1}/\sqrt{\mathbf{1'M}_j\mathbf{1}}$, and $c_j \equiv \sqrt{\mathbf{1'M}_j\mathbf{1}}$. Applying the same procedure to (3.8) as discussed in section 2.3.1, one finds that the global minimum of (3.8) is attained for

$$\mathbf{g}_j = \frac{[\tilde{\mathbf{X}}_j - (\mathbf{Y} - \mathbf{1h}_j')\mathbf{W}_j\mathbf{Q}_j']'\mathbf{M}_j\mathbf{1}}{\mathbf{1'M}_j\mathbf{1}}. \tag{3.9}$$

Substitution of (3.9) in

$$(\tilde{\mathbf{X}}_j - \mathbf{1g}_j')\mathbf{Q}_j - (\mathbf{Y} - \mathbf{1h}_j')\mathbf{W}_j,$$

which is a part of (3.7), gives

$$(\tilde{\mathbf{X}}_j - \frac{\mathbf{11'M}_j[\tilde{\mathbf{X}}_j - (\mathbf{Y} - \mathbf{1h}_j')\mathbf{W}_j\mathbf{Q}_j']}{\mathbf{1'M}_j\mathbf{1}})\mathbf{Q}_j - (\mathbf{Y} - \mathbf{1h}_j')\mathbf{W}_j = \mathbf{J}_j(\tilde{\mathbf{X}}_j\mathbf{Q}_j - \mathbf{YW}_j),$$

with $\mathbf{J}_j = (\mathbf{I} - \frac{\mathbf{11'M}_j}{\mathbf{1'M}_j\mathbf{1}})$.

Hence, defining $\mathbf{C}_j = \mathbf{M}_j\mathbf{J}_j$, and since $\mathbf{C}_j$ is an idempotent matrix, the complete DIMFREE loss function (3.7) can be written as

$$f(\mathbf{Q},\mathbf{W},\mathbf{Y}) = \sum_{j=1}^{n} \mathrm{tr}\,(\tilde{\mathbf{X}}_j\mathbf{Q}_j - \mathbf{YW}_j)'\mathbf{C}_j(\tilde{\mathbf{X}}_j\mathbf{Q}_j - \mathbf{YW}_j). \tag{3.10}$$

Substitution of (3.9) in (3.7) has the nice effect of simultaneously eliminating the $\mathbf{g}_j$ *and* the $\mathbf{h}_j$ from (3.7), which means that model (3.4) can be simplified to

$$\mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j = \mathbf{C}_j\mathbf{Y}\mathbf{W}_j + \mathbf{C}_j\mathbf{E}_j \qquad \text{for } j = 1, 2, ..., n. \tag{3.11}$$

This shows that, just like in GPA, the problem of translations in the DIMFREE model is implicitly solved by centering the non-missing elements in each column of $\tilde{\mathbf{X}}_j$ and the corresponding elements in each column of $\mathbf{Y}$ on the origin of m-dimensional space.

It is always true that $\mathbf{C}_j\mathbf{1} = \mathbf{0}$, and, therefore, that

$$\mathbf{C}_j\tilde{\mathbf{X}}_j = s_j\mathbf{C}_j(\mathbf{X}_j - \mathbf{1}\mathbf{u}_j')\mathbf{R}_j\mathbf{K} = s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j\mathbf{K} - s_j\mathbf{C}_j\mathbf{1}\mathbf{u}_j'\mathbf{R}_j\mathbf{K} = s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j\mathbf{K}, \tag{3.12}$$

where s_j, $\mathbf{u}_j$, and $\mathbf{R}_j$ are the optimal central dilation, translation and orthonormal transformation obtained in GPA, and $\mathbf{K}$ is the orthonormal matrix rotating the GPA solution to its principal components. It follows from (3.12) that nothing changes by using $s_j\mathbf{X}_j\mathbf{R}_j\mathbf{K}$ instead of $s_j(\mathbf{X}_j - \mathbf{1}\mathbf{u}_j')\mathbf{R}_j\mathbf{K}$ when fitting the DIMFREE model on the optimally transformed configurations in GPA. Hence, we define $\tilde{\mathbf{X}}_j = s_j\mathbf{X}_j\mathbf{R}_j\mathbf{K}$ throughout the rest of section 3.3.

3.3.3 Orthonormal transformations

To determine the orthonormal matrices $\mathbf{Q}_j$ minimizing (3.10), define $\mathbf{A}_j = \mathbf{Y}\mathbf{W}_j$ and rewrite (3.10) as

$$f(\mathbf{Q}) = \sum_{j=1}^{n} \text{tr } (\tilde{\mathbf{X}}_j\mathbf{Q}_j - \mathbf{A}_j)'\mathbf{C}_j(\tilde{\mathbf{X}}_j\mathbf{Q}_j - \mathbf{A}_j). \tag{3.13}$$

Considering only one particular $\mathbf{Q}_j$, (3.13) can be written as

$$f(\mathbf{Q}_j) = \text{tr } \mathbf{Q}_j'\tilde{\mathbf{X}}_j'\mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j - 2 \text{ tr } \mathbf{Q}_j'\tilde{\mathbf{X}}_j'\mathbf{C}_j\mathbf{A}_j + b_j, \tag{3.14}$$

where b_j is a term independent of $\mathbf{Q}_j$. Therefore, the minimization of (3.14) is equivalent to the maximization of

$$g(\mathbf{Q}_j) = \text{tr}\ \mathbf{Q}_j'\tilde{\mathbf{X}}_j'\mathbf{C}_j\mathbf{A}_j \tag{3.15}$$

under the constraint that $\mathbf{Q}_j'\mathbf{Q}_j = \mathbf{Q}_j\mathbf{Q}_j' = \mathbf{I}_m$. Letting

$$\tilde{\mathbf{X}}_j'\mathbf{C}_j\mathbf{A}_j = \mathbf{K}_j\mathbf{\Phi}_j\mathbf{L}_j' \tag{3.16}$$

be a singular value decomposition of $\tilde{\mathbf{X}}_j'\mathbf{C}_j\mathbf{A}_j$, the global maximum of (3.15) is found for

$$\mathbf{Q}_j = \mathbf{K}_j\mathbf{L}_j'. \tag{3.17}$$

This analytical solution obviously satisfies the constraint that $\mathbf{Q}_j$ must be orthonormal, and holds regardless of singularity of (3.16). Because (3.17) gives the global minimum of (3.14) with respect to $\mathbf{Q}_j$, determining (3.17) for j = 1, ..., n is guaranteed to yield the global minimum of (3.13) with respect to $\mathbf{Q}$.

3.3.4 Dimension weights

For fixed $\mathbf{Q}$ and $\mathbf{Y}$, the problem discussed in this section is how to minimize

$$f(\mathbf{W}) = \sum_{j=1}^{n} \text{tr}\ (\mathbf{A}_j - \mathbf{B}_j\mathbf{W}_j)'(\mathbf{A}_j - \mathbf{B}_j\mathbf{W}_j), \tag{3.18}$$

where $\mathbf{A}_j \equiv \mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j$ and $\mathbf{B}_j \equiv \mathbf{C}_j\mathbf{Y}$. Considering only one $\mathbf{W}_j$, (3.18) may be written as

$$\begin{aligned} f(\mathbf{W}_j) &= \text{tr}\ \mathbf{A}_j'\mathbf{A}_j + \text{tr}\ \mathbf{W}_j^2(\text{diag}\ \mathbf{B}_j'\mathbf{B}_j) - 2\ \text{tr}\ \mathbf{W}_j(\text{diag}\ \mathbf{A}_j'\mathbf{B}_j) \\ &= d_j + \|\ \mathbf{W}_j(\text{diag}\ \mathbf{B}_j'\mathbf{B}_j)^{1/2} - (\text{diag}\ \mathbf{B}_j'\mathbf{B}_j)^{-1/2}(\text{diag}\ \mathbf{A}_j'\mathbf{B}_j)\ \|^2, \end{aligned} \tag{3.19}$$

where d_j is a constant with respect to $\mathbf{W}_j$. Therefore, the global minimum of (3.19) is attained for

$$\mathbf{W}_j = (\text{diag}\ \mathbf{B}_j'\mathbf{B}_j)^{-1}(\text{diag}\ \mathbf{A}_j'\mathbf{B}_j). \tag{3.20}$$

Because (3.20) gives the global minimum of (3.19), calculating (3.20) for j = 1, ..., n also yields the global minimum of (3.18) with respect to $\mathbf{W}$. In words, the optimal dimension weights in (3.20) are the raw regression weights in the regression equations

$$\mathbf{a}_{kj} = w_{kj}\mathbf{b}_{kj} + \mathbf{e}_{kj}, \quad \text{for } k = 1, 2, ..., m,$$

where $\mathbf{a}_{kj}$ and $\mathbf{b}_{kj}$ are column k of $\mathbf{A}_j$ and $\mathbf{B}_j$, respectively, and w_{kj} is the dimension weight in $\mathbf{W}_j$ corresponding to dimension k.

The last unknown to be determined in (3.10) is the unrestricted group configuration $\mathbf{Y}$. This problem is discussed in the next section.

3.3.5 The centroid configuration Y

For fixed $\mathbf{Q}$ and $\mathbf{W}$, we want to find the minimum of

$$f(\mathbf{Y}) = \sum_{j=1}^{n} \text{tr}\,(\mathbf{A}_j - \mathbf{C}_j\mathbf{Y}\mathbf{W}_j)'(\mathbf{A}_j - \mathbf{C}_j\mathbf{Y}\mathbf{W}_j), \tag{3.21}$$

where $\mathbf{A}_j \equiv \mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j$. Define $\mathbf{a}_j = (\text{vec } \mathbf{A}_j)$, that is, let $\mathbf{a}_j$ be the (mp x 1) vector consisting of the columns of $\mathbf{A}_j$ stacked on top of one another, and let (vec $\mathbf{C}_j\mathbf{Y}\mathbf{W}_j$) denote the (mp x 1) vector which is obtained by stacking the columns of $\mathbf{C}_j\mathbf{Y}\mathbf{W}_j$ on top of one another. Since (see, e.g., Magnus and Neudecker, 1988)

$$(\text{vec } \mathbf{C}_j\mathbf{Y}\mathbf{W}_j) = (\mathbf{W}_j \otimes \mathbf{C}_j)(\text{vec } \mathbf{Y}),$$

where (vec $\mathbf{Y}$) is the (pm x 1) vector containing the columns of $\mathbf{Y}$ stacked on top of one another and $\otimes$ denotes the right Kronecker product, we may rewrite (3.21) as

$$f(\mathbf{y}) = \sum_{j=1}^{n} (\mathbf{a}_j - \mathbf{D}_j\mathbf{y})'(\mathbf{a}_j - \mathbf{D}_j\mathbf{y}), \tag{3.22}$$

where $\mathbf{y} \equiv (\text{vec } \mathbf{Y})$ of order (mp x 1), and $\mathbf{D}_j \equiv (\mathbf{W}_j \otimes \mathbf{C}_j)$ of order (mp x mp). If we finally let the supervector $\mathbf{a}$ of order (nmp x 1) contain the $\mathbf{a}_j$'s stacked on top of one another, and if we collect the matrices $\mathbf{D}_j$ in the supermatrix $\mathbf{D}$ of order (nmp x mp), (3.22) may be written as

$$f(\mathbf{y}) = (\mathbf{a} - \mathbf{D}\mathbf{y})'(\mathbf{a} - \mathbf{D}\mathbf{y}). \tag{3.23}$$

This is the classical univariate multiple regression problem. The solution follows from the normal equations

$$\mathbf{D'Dy} = \mathbf{D'a}, \tag{3.24}$$

which may be re-expressed in terms of the original matrices as

$$[\sum_{j=1}^{n} (\mathbf{W}_j^2 \otimes \mathbf{C}_j)](\text{vec } \mathbf{Y}) = \sum_{j=1}^{n} (\text{vec } \mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j\mathbf{W}_j). \tag{3.25}$$

Hence, the global minimum of (3.21) is attained for

$$\text{vec } \mathbf{Y} = [\sum_{j=1}^{n} (\mathbf{W}_j^2 \otimes \mathbf{C}_j)]^{-} [\sum_{j=1}^{n} (\text{vec } \mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j\mathbf{W}_j)], \tag{3.26}$$

where $[\Sigma (\mathbf{W}_j^2 \otimes \mathbf{C}_j)]^{-}$ is the Moore-Penrose generalized inverse of $\Sigma (\mathbf{W}_j^2 \otimes \mathbf{C}_j)$. The latter matrix of order (pm x pm) is block-diagonal, and the Moore-Penrose generalized inverse of this matrix can therefore be computed for each of the m symmetric blocks of order (p x p) separately. For the determination of the Moore-Penrose inverse of a singular Gramian matrix we refer to section 2.3.2 of chapter 2.

In the special case that all configurations are complete, and assuming that they have been centered on the origin, (3.26) simplifies to

$$\mathbf{Y} = (\sum_{j=1}^{n} \tilde{\mathbf{X}}_j\mathbf{Q}_j\mathbf{W}_j)(\sum_{j=1}^{n} \mathbf{W}_j^2)^{-1}, \tag{3.27}$$

and we only need to determine the proper inverse of an (m x m) diagonal matrix.

3.3.6 The direct approach in DIMFREE

In this section we briefly discuss the consequences of eliminating the centroid configuration **Y** from the loss function for the DIMFREE model by substitution of (3.26) in (3.10). We have investigated this option because we were interested in finding out whether an analogous situation could be created as in the GPA model (see section 2.3.3 of chapter 2), where the elimination of **Z** resulted in a very efficient method for the determination of the unknown orthonormal transformation matrices and scaling factors.

We state, without proof, that substitution of (3.26) in (3.10) yields the following loss function:

$$f(\mathbf{Q},\mathbf{W}) = n - [\text{vec}\,(\sum_{j=1}^{n} \mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j\mathbf{W}_j)]'[\sum_{j=1}^{n} (\mathbf{W}_j^2 \otimes \mathbf{C}_j)]^{-}[\text{vec}\,(\sum_{j=1}^{n} \mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j\mathbf{W}_j)]. \quad (3.28)$$

Although we will not discuss the solution in detail, we note that it is possible to determine the orthonormal matrices $\mathbf{Q}_j$ minimizing (3.28) for fixed $\mathbf{W}$. This solution for the estimation of each $\mathbf{Q}_j$ in (3.28) is closely related to the solution proposed by Mooijaart and Commandeur (1990) for the weighted orthonormal Procrustes problem, and requires an iterative procedure if the configurations are more than two-dimensional. Because the procedure is iterative, it is much more expensive in terms of computation time than the analytical solution for $\mathbf{Q}$ discussed in section 3.3.3. Moreover, in more than two dimensions it is unclear whether this iterative procedure always yields the global minimum of (3.28) with respect to $\mathbf{Q}$, while the procedure described in section 3.3.3 is guaranteed to give the global minimum of (3.10).

As for the dimension weights, we have not been able to solve the problem of minimizing (3.28) with respect to $\mathbf{W}$ for fixed $\mathbf{Q}$ in the case of incomplete configurations. Without missing data, however, this minimization problem can be shown to be an eigenvalue-eigenvector problem for which an analytical solution is available.

Hence, in the general case of three- and higher dimensional incomplete configurations the 'centroid' approach (3.10) is to be preferred to the direct approach (3.28), because the former results in a more efficient procedure for the estimation of the matrices $\mathbf{Q}_j$, and because at present no solution is available for the estimation of the dimension weights $\mathbf{W}_j$ in (3.28) when the configurations are incomplete.

3.3.7 Uniqueness of the DIMFREE solution

The solution for the DIMFREE model is unique up to a weighting of the optimal dimensions weights $\mathbf{W}_j$ together with an inverse weighting of the columns of the optimal centroid $\mathbf{Y}$. This is easily verified by noting that the solution found for this model will always be only one member of the following family of equivalent solutions:

$$f(\mathbf{Q},\mathbf{W},\mathbf{Y}) = \sum_{j=1}^{n} \text{tr}\,(\tilde{\mathbf{X}}_j\mathbf{Q}_j - \mathbf{Y}\mathbf{L}^{-1}\mathbf{L}\mathbf{W}_j)'\mathbf{C}_j(\tilde{\mathbf{X}}_j\mathbf{Q}_j - \mathbf{Y}\mathbf{L}^{-1}\mathbf{L}\mathbf{W}_j), \tag{3.29}$$

where $\mathbf{L}$ is an arbitrary diagonal matrix of order (m x m).

The freedom we have in choosing a particular diagonal matrix $\mathbf{L}$ can be used to select the matrix $\mathbf{L}$ which guarantees the uniqueness of the solution for the DIMFREE model with respect to dimension weighting. We propose to use the following diagonal matrix, which we will denote by $\mathbf{D}$:

$$\mathbf{D} = (\text{diag } \mathbf{Y}'\mathbf{Y})^{1/2}, \tag{3.30}$$

yielding a uniquely weighted DIMFREE solution where

$$\mathbf{Y}^* = \mathbf{Y}\mathbf{D}^{-1} = \mathbf{Y}(\text{diag } \mathbf{Y}'\mathbf{Y})^{-1/2} \tag{3.31}$$

and

$$\mathbf{W}_j^* = \mathbf{D}\mathbf{W}_j = (\text{diag } \mathbf{Y}'\mathbf{Y})^{1/2}\mathbf{W}_j \tag{3.32}$$

for j = 1, 2, ..., n.

That (3.31) and (3.32) guarantee a uniquely weighted solution can be seen as follows. To postmultiply $\mathbf{Y}$ with a diagonal matrix has the effect of rescaling its columns, and $\mathbf{Y}$ is, therefore, unique up to a differential rescaling of its columns. Since postmultiplying $\mathbf{Y}$ with $\mathbf{D}^{-1}$ has the effect of unit normalizing the columns of $\mathbf{Y}$ (i.e. (diag $\mathbf{D}^{-1}\mathbf{Y}'\mathbf{Y}\mathbf{D}^{-1}$) = $\mathbf{I}_m$), it immediately follows that (3.31) yields a uniquely weighted $\mathbf{Y}^*$, and it also follows that (3.32) yields uniquely weighted dimension weights $\mathbf{W}_j^*$.

As we will prove in section 3.5.2, if there are no missing data the choice for (3.30) also guarantees that the sum of squared dimension weights $\mathbf{W}_j^*$ for each j becomes equal to the squared correlation between the elements of $\tilde{\mathbf{X}}_j\mathbf{Q}_j$ and the elements of $\mathbf{Y}\mathbf{W}_j$.

An important property of the DIMFREE solution is that the centroid configuration $\mathbf{Y}$ has optimally rotated dimensions since they are the differentially weighted ones. Analogous to the INDSCAL model, in DIMFREE also it is the model itself that, as it were, dictates which axes are to be used for interpretation. Hence, having uniquely weighted the centroid configuration according to (3.31) and the dimension weights according to (3.32), the DIMFREE solution is unique up to permutations and reflections of the m dimensions. Under these restrictions we are still free to permute the m dimensions such that they are ordered according to the amount of variation accounted for by each dimension. In section 3.5.2 we will give a procedure to determine this order.

Because there is no analytical solution for the unknowns in the DIMFREE model, an alternating least squares algorithm is used for the estimation of the unknown transformation parameters in (3.10).

3.3.8 The algorithm

In the MATCHALS program, the algorithm corresponding to the DIMFREE model consists of the following steps. As input to the algorithm we use the n configurations $\tilde{\mathbf{X}}_j = s_j\mathbf{X}_j\mathbf{R}_j\mathbf{K}$ obtained in GPA.

a) Before starting to iterate, initialize the matrices $\mathbf{Q}_j$ on $\mathbf{Q}_j = \mathbf{I}_m$, and $\mathbf{Y}$ on $\mathbf{Y} = \mathbf{ZK}$ (i.e., the optimal centroid configuration found in GPA rotated to the principal components of the GPA solution).

b) For every j, compute new dimension weights with (3.20).

c) Determine a new orthonormal matrix $\mathbf{Q}_j$ for every j by applying the procedure given in section 3.3.3.

d) Again compute new dimension weights with (3.20) for every j.

e) Calculate a new centroid configuration $\mathbf{Y}$. If there are missing data, update $\mathbf{Y}$ according to (3.26). To determine the Moore-Penrose inverse of the block-diagonal matrix $[\Sigma\ (\mathbf{W}_j^2 \otimes \mathbf{C}_j)]$ of order (pm x pm), compute the Moore-Penrose inverse of each of the m blocks separately. This is more efficient than calculating an inverse of the complete (pm x pm) matrix. If all configurations are complete, however, use (3.27) instead of (3.26) to update $\mathbf{Y}$.

f) Evaluate loss function (3.10). If the difference between the value of the function in this iteration and in the previous iteration is smaller than some predetermined convergence criterion, go to step g). Otherwise, go to step b).

g) Print the history of iterations. Compute $\mathbf{Y}^*$ and $\mathbf{W}_j^*$ using (3.31) and (3.32), respectively, and print the uniquely weighted optimal group configuration $\mathbf{Y}^*$ and the unique dimension weights $\mathbf{W}_j^*$ for each j.

This algorithm is guaranteed to converge, although not necessarily to the global minimum of (3.10). The reason why dimension weights are calculated twice in one iteration (i.e., in steps b) and d) of the algorithm) is that, in comparison with the $\mathbf{Q}_j$ and $\mathbf{Y}$, the estimation of the $\mathbf{W}_j$ is relatively cheap in terms of CPU-time.

We end this section by emphasizing that, since the dimension weights $\mathbf{W}_j^*$ and the group configuration $\mathbf{Y}^*$ are uniquely weighted, the solution for the DIMFREE model is unique up to reflections and permutations of the m dimensions. Because of the unique orientation of the axes of $\mathbf{Y}^*$, just as in the literature on the INDSCAL analysis of (dis)similarities data, it is recommended to use only the uniquely oriented dimensions of $\mathbf{Y}^*$ for interpretational purposes.

3.4 The DIMIDIO model

3.4.1 Introduction

To determine the optimal translations, orthonormal transformations, dimension weights, and centroid configuration in the DIMIDIO model, the following least squares loss function

$$f(\mathbf{G},\mathbf{H},\mathbf{Q},\mathbf{S},\mathbf{W},\mathbf{Y}) = \sum_{j=1}^{n} \| \mathbf{M}_j[(\tilde{\mathbf{X}}_j - \mathbf{1g}_j')\mathbf{Q}_j - (\mathbf{Y} - \mathbf{1h}_j')\mathbf{S}_j\mathbf{W}_j] \|^2 \qquad (3.33)$$

is defined, where **G** and **H** denote the (m x n) matrices in which the translation vectors $\mathbf{g}_j$ and $\mathbf{h}_j$ are collected, respectively, and **Q**, **S**, and **W** denote the supermatrices of order (nm x m) in which the n matrices $\mathbf{Q}_j$, $\mathbf{S}_j$, and $\mathbf{W}_j$ are collected, respectively. Matrix $\tilde{\mathbf{X}}_j$ again denotes an optimally translated, rotated and rescaled configuration $\mathbf{X}_j$ after GPA rotated to the principal components of the GPA solution (i.e., $\tilde{\mathbf{X}}_j = s_j(\mathbf{X}_j - \mathbf{1u}_j')\mathbf{R}_j\mathbf{K}$). In section 3.4.2 we first of all show that the translation vectors $\mathbf{g}_j$ and $\mathbf{h}_j$ can be eliminated from (3.33). In section 3.4.3 we discuss how the minimization of the simplified loss function for the DIMIDIO model with respect to the orthonormal matrices $\mathbf{Q}_j$, dimension weights $\mathbf{W}_j$, and orthonormal matrices $\mathbf{S}_j$ can be reduced to the much more simple problem of minimizing the loss function with respect to one set of parameters only, from which the optimal **Q**, **W**, and **S** can be recovered afterwards. In sections 3.4.4 and 3.4.5 the resulting DIMIDIO loss function is minimized with respect to the only two remaining sets of parameters. In section 3.4.6 the uniqueness properties of the DIMIDIO model are discussed, and in section 3.4.7 an alternating least squares algorithm is presented for the minimization of the loss function corresponding to the DIMIDIO model.

3.4.2 Translations

The problem of minimizing (3.33) with respect to the unrestricted translation vectors **G** for fixed **H**, **Q**, **S**, **W**, and **Y** is so similar to the problem encountered in section 3.3.2 for the DIMFREE model that we state, without proof, that the global minimum of (3.33) with respect to one particular translation vector $\mathbf{g}_j$, that is, of

$$f(\mathbf{g}_j) = \sum_{j=1}^{n} \| \mathbf{M}_j[(\tilde{\mathbf{X}}_j - \mathbf{1g}_j')\mathbf{Q}_j - (\mathbf{Y} - \mathbf{1h}_j')\mathbf{S}_j\mathbf{W}_j] \|^2 \qquad (3.34)$$

is attained for

$$\mathbf{g}_j = \frac{[\tilde{\mathbf{X}}_j - (\mathbf{Y} - \mathbf{1h}_j')\mathbf{S}_j\mathbf{W}_j\mathbf{Q}_j']'\mathbf{M}_j\mathbf{1}}{\mathbf{1}'\mathbf{M}_j\mathbf{1}}. \qquad (3.35)$$

Substitution of (3.35) in (3.33) gives

$$f(\mathbf{Q},\mathbf{S},\mathbf{W},\mathbf{Y}) = \sum_{j=1}^{n} \mathrm{tr}\, (\tilde{\mathbf{X}}_j\mathbf{Q}_j - \mathbf{YS}_j\mathbf{W}_j)'\mathbf{C}_j(\tilde{\mathbf{X}}_j\mathbf{Q}_j - \mathbf{YS}_j\mathbf{W}_j), \qquad (3.36)$$

where $\mathbf{C}_j \equiv \mathbf{M}_j(\mathbf{I} - \mathbf{11}'\mathbf{M}_j/\mathbf{1}'\mathbf{M}_j\mathbf{1})$. Thus, substitution of (3.35) in (3.33) results in the simultaneous elimination of $\mathbf{G}$ *and* $\mathbf{H}$ from (3.33). After the elimination of the translation vectors from (3.33), model (3.5) can be simplified to

$$\mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j = \mathbf{C}_j\mathbf{YS}_j\mathbf{W}_j + \mathbf{C}_j\mathbf{E}_j \qquad \text{for } j = 1, 2, ..., n. \qquad (3.37)$$

Since it follows from $\mathbf{C}_j\mathbf{1} = \mathbf{0}$ that

$$\mathbf{C}_j\tilde{\mathbf{X}}_j = \mathbf{C}_j(s_j\mathbf{X}_j - \mathbf{1u}_j')\mathbf{R}_j\mathbf{K} = s_j\mathbf{C}_j\mathbf{X}_j\mathbf{R}_j\mathbf{K},$$

we may as well fit the DIMIDIO model on the optimally rotated and rescaled configurations after GPA. It is for this reason that we define $\tilde{\mathbf{X}}_j = s_j\mathbf{X}_j\mathbf{R}_j\mathbf{K}$ throughout the remainder of this chapter.

3.4.3 Reducing the estimation of Q, S, and W to one set of parameters

We start this section by noting that, since for orthonormal matrices $\mathbf{Q}_j$

$$f(\mathbf{Q},\mathbf{S},\mathbf{W},\mathbf{Y}) = \sum_{j=1}^{n} \mathrm{tr}\, \mathbf{E}_j'\mathbf{C}_j\mathbf{E}_j = \sum_{j=1}^{n} \mathrm{tr}\, \mathbf{Q}_j\mathbf{E}_j'\mathbf{C}_j\mathbf{E}_j\mathbf{Q}_j',$$

where $\mathbf{E}_j$ is defined in (3.37), the minimization of loss function (3.36) with respect to $\mathbf{Q}$, $\mathbf{S}$, $\mathbf{W}$, and $\mathbf{Y}$ is equivalent to the minimization of

$$f(\mathbf{Q},\mathbf{S},\mathbf{W},\mathbf{Y}) = \sum_{j=1}^{n} \text{tr}\,(\tilde{\mathbf{X}}_j - \mathbf{Y}\mathbf{S}_j\mathbf{W}_j\mathbf{Q}_j')'\mathbf{C}_j(\tilde{\mathbf{X}}_j - \mathbf{Y}\mathbf{S}_j\mathbf{W}_j\mathbf{Q}_j'). \tag{3.38}$$

Letting $\mathbf{B}_j = \mathbf{S}_j\mathbf{W}_j\mathbf{Q}_j'$ of order (m x m), (3.38) may be written as

$$f(\mathbf{B},\mathbf{Y}) = \sum_{j=1}^{n} \text{tr}\,(\tilde{\mathbf{X}}_j - \mathbf{Y}\mathbf{B}_j)'\mathbf{C}_j(\tilde{\mathbf{X}}_j - \mathbf{Y}\mathbf{B}_j), \tag{3.39}$$

where it is assumed that the n matrices $\mathbf{B}_j$ are collected in the supermatrix **B** of order (nm x m). It is important to note that $\mathbf{B}_j$ stands for the matrix product of an orthonormal matrix $\mathbf{S}_j$, a diagonal matrix $\mathbf{W}_j$, and another orthonormal matrix $\mathbf{Q}_j'$. Since, moreover, singular value decomposition can be applied to any matrix, no restrictions have to be imposed on the matrices $\mathbf{B}_j$. If, therefore, we have determined the (unrestricted) matrix **B** which gives the global minimum of (3.39) for fixed **Y**, and given the singular value decompositions

$$\mathbf{B}_j = \mathbf{K}_j\mathbf{\Lambda}_j\mathbf{L}_j' \qquad \text{for } j = 1, \ldots, n, \tag{3.40}$$

then the global minimum of (3.38) with respect to **Q**, **S**, and **W** is obtained by setting $\mathbf{S}_j = \mathbf{K}_j$, $\mathbf{W}_j = \mathbf{\Lambda}_j$, and $\mathbf{Q}_j = \mathbf{L}_j$ for $j = 1, \ldots, n$.

The reformulation of (3.36) in (3.39) considerably simplifies the task of minimizing the DIMIDIO loss function: instead of having to determine four sets of parameters in (3.36) we need to determine only two sets of parameters in (3.39). Once the optimal matrix **B** is obtained, the optimal matrices **S**, **W**, and **Q** can be recovered using (3.40). The following two sections deal with the minimization of (3.39) with respect to the unrestricted matrices **B** and **Y**.

3.4.4 The regression weights B

To minimize (3.39) with respect to unrestricted **B** for fixed **Y**, consider only one $\mathbf{B}_j$ and write (3.39) as

$$f(\mathbf{B}_j) = d_j + \text{tr}\,(\mathbf{A}_j - \mathbf{D}_j\mathbf{B}_j)'(\mathbf{A}_j - \mathbf{D}_j\mathbf{B}_j), \tag{3.41}$$

where $\mathbf{A}_j \equiv \mathbf{C}_j\tilde{\mathbf{X}}_j$, $\mathbf{D}_j \equiv \mathbf{C}_j\mathbf{Y}$, and d_j is a constant with respect to $\mathbf{B}_j$. This is the classical multivariate multiple regression problem with the well-known solution

$$\mathbf{B}_j = (\mathbf{D}_j'\mathbf{D}_j)^{-1}(\mathbf{D}_j'\mathbf{A}_j), \tag{3.42}$$

which, when calculated for j = 1, ..., n, is guaranteed to yield the global minimum of (3.39) with respect to **B**.

It may be interesting to note that, applying the results of sections 3.4.2, 3.4.3 and 3.4.4 to the least squares loss function corresponding to the original model (3.3) of Lingoes and Borg where the centroid configuration was assumed to be fixed (see section 3.2), one finds that the minimization of this loss function has an *analytical* solution.

3.4.5 The centroid configuration Y

For fixed **B**, we want to find the minimum of

$$f(\mathbf{Y}) = \sum_{j=1}^{n} \text{tr}\,(\mathbf{A}_j - \mathbf{C}_j\mathbf{Y}\mathbf{B}_j)'(\mathbf{A}_j - \mathbf{C}_j\mathbf{Y}\mathbf{B}_j), \tag{3.43}$$

where $\mathbf{A}_j \equiv \mathbf{C}_j\tilde{\mathbf{X}}_j$. Applying the same procedure as in section 3.3.5, that is, letting **a** be the (nmp x 1) supervector containing the n vectors (vec $\mathbf{A}_j$) on top of one another, letting **D** be the (nmp x mp) supermatrix containing the n matrices $(\mathbf{B}_j' \otimes \mathbf{C}_j)$, and defining **y** = (vec **Y**), the problem of minimizing (3.40) can be expressed as the classical univariate multiple regression problem:

$$f(\mathbf{y}) = (\mathbf{a} - \mathbf{D}\mathbf{y})'(\mathbf{a} - \mathbf{D}\mathbf{y}). \tag{3.44}$$

Hence, the global minimum of (3.43) is attained for

$$\text{vec}\,\mathbf{Y} = (\mathbf{D}'\mathbf{D})^{-}\mathbf{D}'\mathbf{a} = [\sum_{j=1}^{n} (\mathbf{B}_j\mathbf{B}_j' \otimes \mathbf{C}_j)]^{-} [\text{vec} \sum_{j=1}^{n} \mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{B}_j'], \tag{3.45}$$

where $[\Sigma\, (\mathbf{B}_j\mathbf{B}_j' \otimes \mathbf{C}_j)]^{-}$ is the Moore-Penrose inverse of $\Sigma\, (\mathbf{B}_j\mathbf{B}_j' \otimes \mathbf{C}_j)$. This latter matrix of order (pm x pm) is no longer block-diagonal, and the Moore-Penrose inverse must therefore be determined for the complete matrix.

In the special case that all n configurations are complete, and assuming that they have been centered on the origin, (3.45) simplifies to

$$\mathbf{Y} = (\sum_{j=1}^{n} \tilde{\mathbf{X}}_j\mathbf{B}_j')(\sum_{j=1}^{n} \mathbf{B}_j\mathbf{B}_j')^{-1}, \tag{3.46}$$

and we only need to determine the proper inverse of an (m x m) matrix.

3.4.6 Uniqueness of the DIMIDIO solution

Since both **B** and **Y** in DIMIDIO loss function (3.39) are unrestricted matrices, the value of this function is unchanged by the following transformation:

$$f(\mathbf{B},\mathbf{Y}) = \sum_{j=1}^{n} \mathrm{tr}\,(\tilde{\mathbf{X}}_j - \mathbf{Y}\mathbf{T}^{-1}\mathbf{T}\mathbf{B}_j)'\mathbf{C}_j(\tilde{\mathbf{X}}_j - \mathbf{Y}\mathbf{T}^{-1}\mathbf{T}\mathbf{B}_j), \tag{3.47}$$

where **T** is an arbitrary non-singular matrix of order (m x m). Because of its nice properties, we choose for **T** in (3.47) the special solution

$$\mathbf{G} = \mathbf{H}\mathbf{N}', \tag{3.48}$$

where **H** contains the singular values, and **N** the right singular vectors of the singular value decomposition

$$\mathbf{Y} = \mathbf{M}\mathbf{H}\mathbf{N}'. \tag{3.49}$$

This yields a DIMIDIO solution where

$$\mathbf{Y}^* = \mathbf{Y}\mathbf{G}^{-1} = \mathbf{M}\mathbf{H}\mathbf{N}'(\mathbf{H}\mathbf{N}')^{-1} = \mathbf{M}, \tag{3.50}$$

and

$$\mathbf{B}_j^* = \mathbf{G}\mathbf{B}_j = \mathbf{H}\mathbf{N}'\mathbf{B}_j \tag{3.51}$$

for j = 1, …, n. Since $\mathbf{Y}^* = \mathbf{M}$, the transformed centroid configuration is columnwise orthonormal. Evidently, having defined (3.51) we use the singular value decompositions

$$\mathbf{B}_j^* = \mathbf{K}_j\mathbf{\Phi}_j\mathbf{L}_j' \qquad \text{for } j = 1, \ldots, n \tag{3.52}$$

instead of (3.40) to determine the optimal $\mathbf{S}_j$, $\mathbf{W}_j$, and $\mathbf{Q}_j$ in DIMIDIO loss function (3.36). That is, we let $\mathbf{S}_j = \mathbf{K}_j$ in (3.52), $\mathbf{W}_j = \mathbf{\Phi}_j$ in (3.52), and $\mathbf{Q}_j = \mathbf{L}_j$ in (3.52).

It follows from (3.50), (3.51), and (3.52) that the $\mathbf{YB}_j$ are decomposed into

$$\mathbf{YB}_j = \mathbf{MK}_j\mathbf{\Phi}_j\mathbf{L}'_j = \mathbf{Y}^*\mathbf{S}_j\mathbf{W}_j\mathbf{Q}'_j, \qquad \text{for } j = 1, \ldots, n, \tag{3.53}$$

which, by letting $\mathbf{V}_j = \mathbf{Y}^*\mathbf{S}_j = \mathbf{MK}_j$, may be written as

$$\mathbf{YB}_j = \mathbf{V}_j\mathbf{\Phi}_j\mathbf{L}'_j, \qquad \text{for } j = 1, \ldots, n. \tag{3.54}$$

Since $\mathbf{V}_j$ is an orthonormal matrix, being the product of orthonormal matrices $\mathbf{M}$ and $\mathbf{K}_j$, (3.54) is a singular value decomposition of $\mathbf{YB}_j$. Therefore, the matrices $\mathbf{W}_j$ and $\mathbf{Q}_j$ defined by (3.52) have the nice property that they are unique up to reflections. The same holds for the matrix products $\mathbf{Y}^*\mathbf{S}_j = \mathbf{MK}_j$. Only one indeterminacy remains: the matrices $\mathbf{Y}^* = \mathbf{M}$ and $\mathbf{S}_j = \mathbf{K}_j$ themselves are only determined up to a rotation, because we are still free to write (3.53) as

$$\mathbf{YB}_j = \mathbf{MP'PK}_j\mathbf{\Phi}_j\mathbf{L}'_j = \mathbf{Y}^*\mathbf{P'PS}_j\mathbf{W}_j\mathbf{Q}'_j, \qquad \text{for } j = 1, \ldots, n, \tag{3.55}$$

where $\mathbf{P}$ is an arbitrary orthonormal matrix of order (m x m). In other words, the matrix product $\mathbf{Y}^*\mathbf{P}'$ just defines another orthonormal basis of the raw $\mathbf{Y}$.

The interpretation of the DIMIDIO model does not require a unique orientation of the centroid configuration, however, because it is the dimensions of the $\mathbf{Y}^*\mathbf{S}_j$ that are differentially being weighted, and the procedure proposed in this section guarantees the uniqueness of the latter matrix products.

As we will prove in section 3.5.3, another advantage of the choice for (3.48) is that, if all configurations are complete, the sum of squared dimension weights $\mathbf{W}_j$ for each j is equal to the squared correlation between the elements of $\tilde{\mathbf{X}}_j\mathbf{Q}_j$ and the elements of $\mathbf{Y}^*\mathbf{S}_j\mathbf{W}_j$.

In the following section we present an alternating least squares algorithm for the estimation of the unknown parameters in the DIMIDIO model.

3.4.7 The algorithm

In the MATCHALS program, the algorithm corresponding to the DIMIDIO model consists of the following steps.

a) Before starting to iterate, initialize **Y** on **ZK** (i.e., the optimal centroid configuration in GPA rotated to the principal components of the GPA solution).
b) Compute new regression weights with (3.42) for every j.
c) Calculate a new centroid configuration **Y**. If the configurations are incomplete, formula (3.45) must be used to update **Y**. If, on the other hand, all configurations are complete, use (3.46) instead of (3.45) since this is much cheaper in terms of CPU-time.
d) Evaluate loss function (3.39). If the difference between the value of the function in this outer iteration and in the previous iteration is smaller than some convergence criterion, go to step e). Otherwise, go to step b).
e) Print the history of iterations. Determine the singular value decomposition of the optimal **Y**, and compute $\mathbf{Y}^*$ and $\mathbf{B}_j^*$ for each j as defined in (3.50) and (3.51), respectively. Then determine the singular value decompositions $\mathbf{B}_j^* = \mathbf{K}_j\mathbf{\Lambda}_j\mathbf{L}'_j$ for each j, and set $\mathbf{S}_j = \mathbf{K}_j$, $\mathbf{W}_j = \mathbf{\Lambda}_j$, and $\mathbf{Q}_j = \mathbf{L}_j$ for each j. Print $\mathbf{Y}^*$, and the $\mathbf{S}_j$, $\mathbf{W}_j$, and $\mathbf{Q}_j$.

This algorithm must converge although it can not be proved that it will necessarily find the global minimum of (3.36). Since in steps b) and c) both **B** and **Y** are updated by the computation of regression weights, the iterative part of the DIMIDIO algorithm is entirely based on multiple regression.

The computation of the Moore-Penrose inverse of the (pm x pm) matrix needed in the case of missing data in step c) of the algorithm is quite expensive. If the number of dimensions *and* the number of stimuli is large, in practice the problem of calculating the Moore-Penrose inverse in (3.45) may well prove to be impossible to cope with. This problem does not arise with complete configurations, of course, because then (3.46) can be used to update **Y**, which only involves the computation of a proper inverse of order (m x m). Still, we emphasize that fitting the DIMIDIO model on large incomplete configurations can become impractical due to computational limits that are easily reached in step c) of the algorithm.

As we already noted in section 3.4.6, the procedure described in step e) of the algorithm guarantees the uniqueness of the DIMIDIO solution up to a rotation of $\mathbf{Y}^*$ together with an inverse rotation of the matrices $\mathbf{S}_j$.

3.5 Analysis of variation in the dimension weighting models

3.5.1 Introduction

In the following sections we will show that the total sum of squares of the n configurations can be partitioned in two components for the two dimension weighting models discussed in this chapter. These two components are again decomposed in three ways in order to allow one to assess the relative contribution of each individual configuration, of each individual stimulus, and of each dimension to the total solution. We first discuss the analysis of variation for the DIMFREE model (3.4) in section 3.5.2, and the analysis of variation for the DIMIDIO model (3.5) is the subject of section 3.5.3.

3.5.2 Partitioning of the sums of squares in DIMFREE

The linear model (3.11) underlying DIMFREE loss function (3.10) is (see section 3.3.2)

$$\mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j = \mathbf{C}_j\mathbf{Y}\mathbf{W}_j + \mathbf{C}_j\mathbf{E}_j \qquad \text{for } j = 1, 2, \ldots, n,$$

from which it follows that

$$\text{tr } \mathbf{Q}_j'\tilde{\mathbf{X}}_j'\mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j = \text{tr } (\mathbf{Y}\mathbf{W}_j + \mathbf{E}_j)'\mathbf{C}_j(\mathbf{Y}\mathbf{W}_j + \mathbf{E}_j), \tag{3.56}$$

and therefore that

$$\text{tr } \tilde{\mathbf{X}}_j'\mathbf{C}_j\tilde{\mathbf{X}}_j = \text{tr } \mathbf{W}_j\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{W}_j + \text{tr } \mathbf{E}_j'\mathbf{C}_j\mathbf{E}_j + 2 \text{ tr } \mathbf{W}_j\mathbf{Y}'\mathbf{C}_j\mathbf{E}_j. \tag{3.57}$$

Because (tr $\mathbf{Y}'\mathbf{C}_j\mathbf{E}_j$) = 0, due to the orthogonality of residual and predictor space in regression analysis, (3.57) may be written as

$$\text{tr } \tilde{\mathbf{X}}_j'\mathbf{C}_j\tilde{\mathbf{X}}_j = \text{tr } \mathbf{W}_j\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{W}_j + \text{tr } \mathbf{E}_j'\mathbf{C}_j\mathbf{E}_j, \qquad \text{for } j = 1, \ldots, n, \tag{3.58}$$

from which it immediately follows that

$$0 \leq (\text{tr}\ \mathbf{W}_j\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{W}_j)/(\text{tr}\ \tilde{\mathbf{X}}'_j\mathbf{C}_j\tilde{\mathbf{X}}_j) \leq 1, \qquad \text{for } j = 1, ..., n, \tag{3.59}$$

showing that (3.59) directly yields a measure of fit for each configuration j in the DIMFREE solution. It also follows from (3.58) that

$$\sum_{j=1}^{n} \text{tr}\ \tilde{\mathbf{X}}'_j\mathbf{C}_j\tilde{\mathbf{X}}_j = \sum_{j=1}^{n} \text{tr}\ \mathbf{W}_j\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{W}_j + \sum_{j=1}^{n} \text{tr}\ \mathbf{E}'_j\mathbf{C}_j\mathbf{E}_j,$$

and therefore that

$$n = \sum_{j=1}^{n} \text{tr}\ \mathbf{W}_j\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{W}_j + f(\mathbf{Q},\mathbf{W},\mathbf{Y}), \tag{3.60}$$

with $f(\mathbf{Q},\mathbf{W},\mathbf{Y})$ as in (3.10), and thus that

$$0 \leq (1/n) \sum_{j=1}^{n} \text{tr}\ \mathbf{W}_j\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{W}_j \leq 1. \tag{3.61}$$

Hence, (3.61) is perfectly suited as a measure of total fit.

The measure of fit in (3.59) is equal to the squared correlation between the elements of $\mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j$ and the elements of $\mathbf{C}_j\mathbf{Y}\mathbf{W}_j$. This simple relation holds, because

$$r^2(\mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j,\mathbf{C}_j\mathbf{Y}\mathbf{W}_j) = \frac{(\text{tr}\ \mathbf{W}_j\mathbf{Y}'\mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j)^2}{(\text{tr}\ \tilde{\mathbf{X}}'_j\mathbf{C}_j\tilde{\mathbf{X}}_j)(\text{tr}\ \mathbf{W}_j\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{W}_j)},$$

$$= (\text{tr}\ \mathbf{W}_j\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{W}_j)/(\text{tr}\ \tilde{\mathbf{X}}'_j\mathbf{C}_j\tilde{\mathbf{X}}_j). \tag{3.62}$$

In section 3.3.7 we proposed to normalize the raw dimension weights in the DIMFREE model according to (3.32), that is:

$$\mathbf{W}_j^* = \mathbf{D}\mathbf{W}_j = (\text{diag}\ \mathbf{Y}'\mathbf{Y})^{1/2}\mathbf{W}_j.$$

In the case that all configurations are complete, and assuming that they have been centered on the origin, it is true that

$$\text{tr}\ [\mathbf{W}_j^*/(\sqrt{\text{tr}\ \tilde{\mathbf{X}}'_j\tilde{\mathbf{X}}_j})]^2 = r^2(\tilde{\mathbf{X}}_j\mathbf{Q}_j,\mathbf{Y}\mathbf{W}_j). \tag{3.63}$$

Table 3.1 Contribution of individual configurations in the DIMFREE model.

configuration j	SS_{fit}	$SS_{residual}$	SS_{total}
1	$\text{tr } \mathbf{W}_1\mathbf{Y}'\mathbf{C}_1\mathbf{Y}\mathbf{W}_1$	$\text{tr } \mathbf{E}_1'\mathbf{C}_1\mathbf{E}_1$	$\text{tr } \tilde{\mathbf{X}}_1'\mathbf{C}_1\tilde{\mathbf{X}}_1$
2	$\text{tr } \mathbf{W}_2\mathbf{Y}'\mathbf{C}_2\mathbf{Y}\mathbf{W}_2$	$\text{tr } \mathbf{E}_2'\mathbf{C}_2\mathbf{E}_2$	$\text{tr } \tilde{\mathbf{X}}_2'\mathbf{C}_2\tilde{\mathbf{X}}_2$
⋮	⋮	⋮	⋮
n	$\text{tr } \mathbf{W}_n\mathbf{Y}'\mathbf{C}_n\mathbf{Y}\mathbf{W}_n$	$\text{tr } \mathbf{E}_n'\mathbf{C}_n\mathbf{E}_n$	$\text{tr } \tilde{\mathbf{X}}_n'\mathbf{C}_n\tilde{\mathbf{X}}_n$
$\sum_{j=1}^{n}$	$\sum_{j=1}^{n} \text{tr } \mathbf{W}_j\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{W}_j$	$\sum_{j=1}^{n} \text{tr } \mathbf{E}_j'\mathbf{C}_j\mathbf{E}_j$	n

Identity (3.63) follows from substitution of (3.32) in the left side term of (3.63):

$$\text{tr } [\mathbf{W}_j^*/(\sqrt{\text{tr } \tilde{\mathbf{X}}'_j\tilde{\mathbf{X}}_j})]^2 = (\text{tr } \mathbf{W}_j^*)^2/(\text{tr } \tilde{\mathbf{X}}'_j\tilde{\mathbf{X}}_j) = (\text{tr } \mathbf{W}_j\mathbf{D}^2\mathbf{W}_j)/(\text{tr } \tilde{\mathbf{X}}'_j\tilde{\mathbf{X}}_j) =$$

$$(\text{tr } \mathbf{W}_j(\text{diag } \mathbf{Y}'\mathbf{Y})\mathbf{W}_j)/(\text{tr } \tilde{\mathbf{X}}'_j\tilde{\mathbf{X}}_j) = (\text{tr } \mathbf{W}_j\mathbf{Y}'\mathbf{Y}\mathbf{W}_j)/(\text{tr } \tilde{\mathbf{X}}'_j\tilde{\mathbf{X}}_j) = r^2(\tilde{\mathbf{X}}_j\mathbf{Q}_j, \mathbf{Y}\mathbf{W}_j).$$

In words (3.63) expresses that, without missing data, the squared norm of the normalized dimension weights contained in $\mathbf{W}_j^*/(\sqrt{\text{tr } \tilde{\mathbf{X}}'_j\tilde{\mathbf{X}}_j})$ directly reflects the amount of variation in configuration j that is accounted for by the DIMFREE model.

We use the above results to give a further decomposition of the sums of squares in (3.60) allowing one to assess the relative contribution of individual configurations, stimuli and dimensions to the DIMFREE solution. In Table 3.1 the decomposition of (3.60) with respect to the individual configurations is given. This table is, of course, nothing more than (3.58) in tabulated form.

Table 3.2 shows how to assess the relative contribution of each individual stimulus to the DIMFREE solution. It is important to note that in this model no independent sums of squares can be obtained for each stimulus separately, both with and without missing data. Therefore, we propose to calculate the amount of variation accounted for by each stimulus according to the expression given in the column titled 'SS_{fit}' in Table 3.2, since this guarantees that SS_{fit} and $SS_{residual}$ add up to SS_{total} for each stimulus i, and that Σ SS_{fit} is equal to the total sum of squares in (3.61). But it must be kept in mind that these decompositions do not result in independent sums of squares.

Table 3.2 Contribution of individual stimuli in the DIMFREE model.

stimulus i	SS_{fit}	$SS_{residual}$	SS_{total}
1	a_{11}*	b_{11}**	d_{11}***
2	a_{22}*	b_{22}**	d_{22}***
⋮	⋮	⋮	⋮
p	a_{pp}*	b_{pp}**	d_{pp}***
$\sum_{i=1}^{p}$	$\sum_{j=1}^{n} \text{tr}\, \mathbf{W}_j\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{W}_j$	$\sum_{j=1}^{n} \text{tr}\, \mathbf{E}_j'\mathbf{C}_j\mathbf{E}_j$	n

* a_{ii} is diagonal element ii of $\sum_{j=1}^{n} \{2\, \mathbf{C}_j\mathbf{Y}\mathbf{W}_j(\mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j)' - \mathbf{C}_j\mathbf{Y}\mathbf{W}_j(\mathbf{C}_j\mathbf{Y}\mathbf{W}_j)'\}$

** b_{ii} is diagonal element ii of $\sum_{j=1}^{n} \mathbf{C}_j\mathbf{E}_j(\mathbf{C}_j\mathbf{E}_j)'$

*** d_{ii} is diagonal element ii of $\sum_{j=1}^{n} \mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j(\mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j)'$

Table 3.3 Contribution of individual dimensions in the DIMFREE model.

dimension k	SS_{fit}	$SS_{residual}$	SS_{total}
1	e_{11}*	f_{11}**	g_{11}***
2	e_{22}*	f_{22}**	g_{22}***
⋮	⋮	⋮	⋮
m	e_{mm}*	f_{mm}**	g_{mm}***
$\sum_{k=1}^{m}$	$\sum_{j=1}^{n} \text{tr}\, \mathbf{W}_j\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{W}_j$	$\sum_{j=1}^{n} \text{tr}\, \mathbf{E}_j'\mathbf{C}_j\mathbf{E}_j$	n

* e_{kk} is diagonal element kk of $\sum_{j=1}^{n} \mathbf{W}_j\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{W}_j$

** f_{kk} is diagonal element kk of $\sum_{j=1}^{n} \mathbf{E}_j'\mathbf{C}_j\mathbf{E}_j$

*** g_{kk} is diagonal element kk of $\sum_{j=1}^{n} \mathbf{Q}_j'\tilde{\mathbf{X}}_j'\mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j$

The partitioning of the sums of squares in (3.60) for each dimension separately is given in Table 3.3. This decomposition yields independent components both with and without missing data, and the information in this table can be used to permute the m dimensions such that they become ordered according to the amount of variation accounted for by each dimension.

3.5.3 Partitioning of the sums of squares in DIMIDIO

The linear model (3.37) underlying DIMIDIO loss function (3.36) is (see section 3.4.3)

$$\mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j = \mathbf{C}_j\mathbf{Y}\mathbf{S}_j\mathbf{W}_j + \mathbf{C}_j\mathbf{E}_j, \qquad \text{for } j = 1, 2, ..., n,$$

from which it follows that

$$\text{tr } \tilde{\mathbf{X}}'_j\mathbf{C}_j\tilde{\mathbf{X}}_j = \text{tr } (\mathbf{Y}\mathbf{S}_j\mathbf{W}_j + \mathbf{E}_j)'\mathbf{C}_j(\mathbf{Y}\mathbf{S}_j\mathbf{W}_j + \mathbf{E}_j), \text{ for } j = 1, ..., n. \qquad (3.64)$$

Expanding the trace on the right side of (3.64) gives

$$\text{tr } \mathbf{W}_j\mathbf{S}'_j\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{S}_j\mathbf{W}_j + \text{tr } \mathbf{E}'_j\mathbf{C}_j\mathbf{E}_j + 2 \text{ tr } \mathbf{W}_j\mathbf{S}'_j\mathbf{Y}'\mathbf{C}_j\mathbf{E}_j =$$

$$\text{tr } \mathbf{Q}_j\mathbf{W}_j\mathbf{S}'_j\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{S}_j\mathbf{W}_j\mathbf{Q}'_j + \text{tr } \mathbf{E}'_j\mathbf{C}_j\mathbf{E}_j + 2 \text{ tr } \mathbf{Q}_j\mathbf{W}_j\mathbf{S}'_j\mathbf{Y}'\mathbf{C}_j\mathbf{E}_j\mathbf{Q}'_j =$$

$$\text{tr } \mathbf{B}'_j\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{B}_j + \text{tr } \mathbf{E}'_j\mathbf{C}_j\mathbf{E}_j + 2 \text{ tr } \mathbf{B}'_j\mathbf{Y}'\mathbf{C}_j\mathbf{E}_j\mathbf{Q}'_j,$$

where $\mathbf{B}_j \equiv \mathbf{S}_j\mathbf{W}_j\mathbf{Q}'_j$. Also, the optimal matrix $\mathbf{B}_j$ contains the regression weights of the regression of $\mathbf{C}_j\tilde{\mathbf{X}}_j$ on $\mathbf{C}_j\mathbf{Y}$ (see section 3.4.4), so that the sum of cross products (tr $\mathbf{B}'_j\mathbf{Y}'\mathbf{C}_j\mathbf{E}_j\mathbf{Q}'_j$) is zero. Therefore (3.64) may be written as

$$\text{tr } \tilde{\mathbf{X}}'_j\mathbf{C}_j\tilde{\mathbf{X}}_j = \text{tr } \mathbf{B}'_j\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{B}_j + \text{tr } \mathbf{E}'_j\mathbf{C}_j\mathbf{E}_j, \qquad \text{for } j = 1, ..., n. \qquad (3.65)$$

It immediately follows that

$$0 \le (\text{tr } \mathbf{B}'_j\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{B}_j)/(\text{tr } \tilde{\mathbf{X}}'_j\mathbf{C}_j\tilde{\mathbf{X}}_j) \le 1, \qquad (3.66)$$

and

$$(\text{tr } \mathbf{B}_j'\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{B}_j)/(\text{tr } \tilde{\mathbf{X}}'_j\mathbf{C}_j\tilde{\mathbf{X}}_j) = r^2(\mathbf{C}_j\tilde{\mathbf{X}}_j\mathbf{Q}_j, \mathbf{C}_j\mathbf{Y}^*\mathbf{S}_j\mathbf{W}_j) \tag{3.67}$$

is a measure of fit for each configuration j in DIMIDIO. If all configurations are complete, and assuming that they have been centered on the origin, it is also true that

$$(\text{tr } \mathbf{B}_j'\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{B}_j)/(\text{tr } \tilde{\mathbf{X}}'_j\mathbf{C}_j\tilde{\mathbf{X}}_j) = (\text{tr } \mathbf{B}_j'\mathbf{Y}'\mathbf{Y}\mathbf{B}_j)/(\text{tr } \tilde{\mathbf{X}}'_j\tilde{\mathbf{X}}_j) =$$

$$(\text{tr } \mathbf{Q}_j\mathbf{W}_j\mathbf{S}_j'\mathbf{Y}^{*\prime}\mathbf{Y}^*\mathbf{S}_j\mathbf{W}_j\mathbf{Q}_j')/(\text{tr } \tilde{\mathbf{X}}'_j\tilde{\mathbf{X}}_j) = \text{tr } [\mathbf{W}_j/(\sqrt{\text{tr } \tilde{\mathbf{X}}'_j\tilde{\mathbf{X}}_j})]^2, \tag{3.68}$$

due to the fact that $\mathbf{Y}^*$ is columnwise orthonormal. In words, (3.68) expresses that, without missing data, the squared norm of the dimension weights $\mathbf{W}_j$ corrected for the size of the corresponding configuration $\tilde{\mathbf{X}}_j$ is equal to the fit of configuration j in the DIMIDIO solution. It also follows from (3.65) that

$$n = \sum_{j=1}^{n} \text{tr } \mathbf{B}_j'\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{B}_j + f(\mathbf{B},\mathbf{Y}), \tag{3.69}$$

with $f(\mathbf{B},\mathbf{Y})$ as in (3.39), and therefore that

$$0 \leq (1/n) \sum_{j=1}^{n} \text{tr } \mathbf{B}_j'\mathbf{Y}'\mathbf{C}_j\mathbf{Y}\mathbf{B}_j \leq 1,$$

is a measure of total fit in the DIMIDIO model.

For a further decomposition of the sums of squares in (3.69) with respect to configurations, stimuli, and dimensions we refer to Tables 3.1, 3.2, and 3.3, respectively, where in all tables one should now read $\mathbf{Y}^*\mathbf{S}_j$ instead of $\mathbf{Y}$.

The MATCHALS program computes and prints all decompositions discussed in section 3.5.

3.6 Illustrations

In this section several examples of the matching of configurations with the dimension weighting models are presented. The algorithms used for the analyses were all programmed in APL, and we used a convergence criterion of 1E-7 throughout. In section 3.6.1 the results of two analyses of a constructed data set with the DIMFREE model are discussed, and in section 3.6.2 the results are given of the DIMIDIO analysis of a constructed data set.

3.6.1 DIMFREE analysis of a constructed data set

The data discussed in this section were constructed in such a way that a perfect solution exists when the admissible transformations are those corresponding to the DIMFREE model (3.4). The 'mould' of this data set is a configuration whose coordinates are given in matrix **A** in Table 3.4. This matrix contains the coordinates of

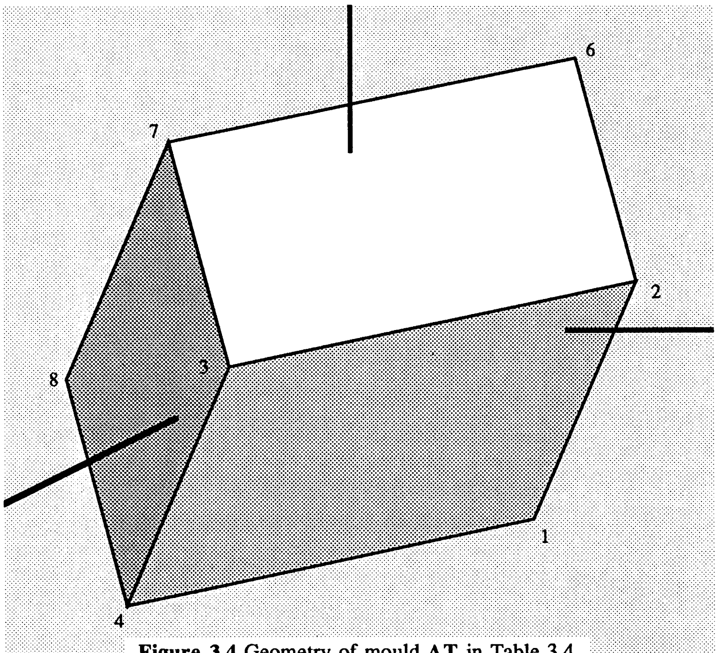


Figure 3.4 Geometry of mould **AT** in Table 3.4.

Table 3.4 Matrices, vectors and central dilations used in the construction of data in Table 3.5.

$$\mathbf{A} = \begin{bmatrix} -0.5 & -0.5 & -0.5 \\ 0.5 & -0.5 & -0.5 \\ 0.5 & 0.5 & -0.5 \\ -0.5 & 0.5 & -0.5 \\ -0.5 & -0.5 & 0.5 \\ 0.5 & -0.5 & 0.5 \\ 0.5 & 0.5 & 0.5 \\ -0.5 & 0.5 & 0.5 \end{bmatrix} \qquad \mathbf{T} = \begin{bmatrix} 0.5000 & 0.1464 & 0.8536 \\ -0.5000 & 0.8536 & 0.1464 \\ -0.7071 & -0.5000 & 0.5000 \end{bmatrix}$$

$$\mathbf{W}_1 = \begin{bmatrix} 4.0 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 2.0 \end{bmatrix} \quad \mathbf{u}_1 = \begin{bmatrix} 2.0 \\ 0.5 \\ 0.8 \end{bmatrix} \quad \mathbf{R}_1 = \begin{bmatrix} 0.3299 & 0.8949 & -0.3004 \\ -0.0501 & 0.3343 & 0.9411 \\ 0.9427 & -0.2955 & 0.1552 \end{bmatrix}$$

$s_1 = 3.0$

$$\mathbf{W}_2 = \begin{bmatrix} 0.8 & 0 & 0 \\ 0 & 2.5 & 0 \\ 0 & 0 & 1.0 \end{bmatrix} \quad \mathbf{u}_2 = \begin{bmatrix} -2.0 \\ -0.4 \\ -1.5 \end{bmatrix} \quad \mathbf{R}_2 = \begin{bmatrix} -0.1065 & 0.1104 & 0.9882 \\ 0.5038 & 0.8628 & -0.0421 \\ 0.8572 & -0.4934 & 0.1475 \end{bmatrix}$$

$s_2 = 0.5$

$$\mathbf{W}_3 = \begin{bmatrix} 1.5 & 0 & 0 \\ 0 & 3.0 & 0 \\ 0 & 0 & 0.2 \end{bmatrix} \quad \mathbf{u}_3 = \begin{bmatrix} 0.2 \\ -1.0 \\ 3.0 \end{bmatrix} \quad \mathbf{R}_3 = \begin{bmatrix} 0.3231 & 0.9355 & 0.1429 \\ 0.7710 & -0.3477 & 0.5336 \\ -0.5489 & 0.0623 & 0.8336 \end{bmatrix}$$

$s_3 = 1.0$

$$\mathbf{W}_4 = \begin{bmatrix} 0.4 & 0 & 0 \\ 0 & 1.0 & 0 \\ 0 & 0 & 4.5 \end{bmatrix} \quad \mathbf{u}_4 = \begin{bmatrix} 4.0 \\ -2.0 \\ 0.0 \end{bmatrix} \quad \mathbf{R}_4 = \begin{bmatrix} 0.9810 & 0.0457 & -0.1876 \\ 0.1906 & -0.3912 & 0.9004 \\ 0.0323 & 0.9193 & 0.3923 \end{bmatrix}$$

$s_4 = 0.8$

the vertices of a cube with edges parallel to the three coordinate axes. As a first step in the construction of artificial data, matrix **A** was rotated by the orthonormal matrix **T** given in Table 3.4. The position in space of the rotated cube **AT** is illustrated in Figure 3.4. The horizontal axis in this figure represents the first dimension, while the

Table 3.5 Constructed data for analysis with DIMFREE model.

$$\mathbf{X}_1 = \begin{bmatrix} -6.9903 & -0.1608 & -2.3074 \\ -0.1939 & 3.7690 & -3.1081 \\ -1.4094 & -1.4322 & 0.0353 \\ -8.2058 & -5.3621 & 0.8360 \\ -6.9244 & -8.8918 & 0.0008 \\ -0.1280 & -4.9619 & -0.7999 \\ -1.3436 & -10.1632 & 2.3435 \\ -8.1400 & -14.0931 & 3.1442 \end{bmatrix} \quad \mathbf{X}_2 = \begin{bmatrix} 0.1432 & -0.1561 & 1.1880 \\ 0.5800 & -0.1867 & 1.4408 \\ 1.2016 & 0.6756 & 1.2091 \\ 0.7649 & 0.7062 & 0.9562 \\ 0.0728 & -0.8499 & 0.9716 \\ 0.5095 & -0.8805 & 1.2245 \\ 1.1312 & -0.0181 & 0.9928 \\ 0.6944 & 0.0124 & 0.7399 \end{bmatrix}$$

$$\mathbf{X}_3 = \begin{bmatrix} 2.0284 & 0.0259 & -2.4452 \\ -2.5157 & 0.5854 & -1.9613 \\ 4.2315 & -1.0048 & -0.6777 \\ 3.7441 & -1.5643 & -1.1616 \\ 0.4744 & -0.4385 & -3.3138 \\ 0.9617 & 0.1210 & -2.8299 \\ 2.6775 & -1.4693 & -1.5463 \\ 2.1901 & -2.0288 & -2.0301 \end{bmatrix} \quad \mathbf{X}_4 = \begin{bmatrix} -2.8487 & -3.1709 & 0.7805 \\ -2.5702 & -0.3845 & 2.0615 \\ -2.5801 & -0.1743 & 2.9133 \\ -2.8585 & -2.9606 & 1.6322 \\ -3.0888 & -1.3700 & 1.1690 \\ -2.8104 & 1.4164 & 2.4500 \\ -2.8202 & 1.6267 & 3.3017 \\ -3.0986 & -1.1597 & 2.0207 \end{bmatrix}$$

vertical axis is the third dimension. Four configurations were constructed from this mould by differentially weighting the dimensions of **AT**. Then, as a final step (to make the confusion complete, as it were), the four identically rotated but differentially weighted configurations were again idiosyncratically translated, rotated and centrally dilated. Table 3.4 contains all transformation parameters that we used to construct this data set. By calculating $\mathbf{X}_j = s_j(\mathbf{ATW}_j - \mathbf{1u}_j')\mathbf{R}_j$ for j = 1, ..., 4 we obtained the configurations given in Table 3.5.

The four complete configurations in Table 3.5 were first submitted to a generalized Procrustes analysis. The corresponding algorithm converged in five iterations, yielding a loss of 1.2928. Since the configurations are always normalized on $\Sigma\ \mathbf{X}_j'\mathbf{C}_j\mathbf{X}_j = n$, this means that the GPA model accounts for (100 x (4 – 1.2928)/4) = 68% of the sum of squares of the four configurations.

Thereafter, the optimally rotated and rescaled configurations $s_j\mathbf{X}_j\mathbf{R}_j\mathbf{K}$ rotated to the principal components of the GPA solution, and the optimal centroid configuration **ZK**, were used as input for the DIMFREE algorithm described in section 3.3.8. This algorithm converged in 3 iterations yielding a loss of 0.0000. As expected, the DIMFREE model accounts for all of the total variation in the four configurations. The results of the DIMFREE analysis are given in Table 3.6.

Table 3.6 Results of DIMFREE analysis of data in Table 3.5.

History of iterations		TOTAL LOSS		
Iteration number	new $\mathbf{W}_j$	new $\mathbf{Q}_j$	new $\mathbf{W}_j$	new $\mathbf{Y}$
1	0.00000139	0.00000092	0.00000092	0.00000038
2	0.00000038	0.00000016	0.00000016	0.00000006
3	0.00000006	0.00000003	0.00000003	0.00000001

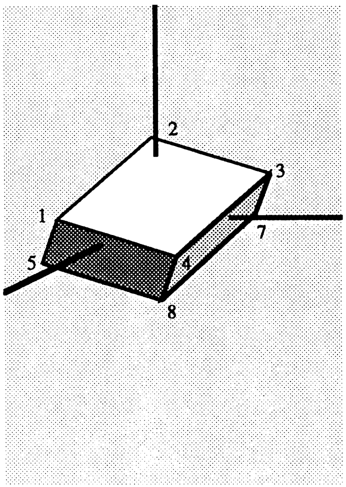
$$\mathbf{Y} = \begin{bmatrix} -0.0993 & 0.2260 & 0.1060 \\ -0.0411 & -0.0312 & 0.2559 \\ 0.2978 & -0.0753 & 0.1060 \\ 0.2397 & 0.1819 & -0.0439 \\ -0.2978 & 0.0753 & -0.1060 \\ -0.2397 & -0.1819 & 0.0439 \\ 0.0993 & -0.2260 & -0.1060 \\ 0.0411 & 0.0312 & -0.2559 \end{bmatrix} \qquad \mathbf{Y}^* = \begin{bmatrix} -0.1768 & 0.5303 & 0.2500 \\ -0.0732 & -0.0733 & 0.6035 \\ 0.5303 & -0.1768 & 0.2500 \\ 0.4268 & 0.4268 & -0.1035 \\ -0.5303 & 0.1768 & -0.2500 \\ -0.4268 & -0.4268 & 0.1035 \\ 0.1768 & -0.5303 & -0.2500 \\ 0.0732 & 0.0733 & -0.6035 \end{bmatrix}$$

$$\mathbf{Q}_1 = \begin{bmatrix} 1.0000 & 0.0001 & 0.0000 \\ -0.0001 & 1.0000 & -0.0008 \\ 0.0000 & 0.0008 & 1.0000 \end{bmatrix} \qquad \mathbf{Q}_2 = \begin{bmatrix} 1.0000 & 0.0001 & 0.0000 \\ -0.0001 & 1.0000 & -0.0007 \\ 0.0000 & 0.0007 & 1.0000 \end{bmatrix}$$

$$\mathbf{Q}_3 = \begin{bmatrix} 1.0000 & 0.0001 & 0.0000 \\ -0.0001 & 1.0000 & -0.0008 \\ 0.0000 & 0.0008 & 1.0000 \end{bmatrix} \qquad \mathbf{Q}_4 = \begin{bmatrix} 1.0000 & 0.0001 & 0.0000 \\ -0.0001 & 1.0000 & -0.0008 \\ 0.0000 & 0.0008 & 1.0000 \end{bmatrix}$$

	raw dimension weights $\mathbf{W}_j$				unique dimension weights $\mathbf{W}_j^*$			
	configuration				configuration			
dimension	1	2	3	4	1	2	3	4
1	0.1838	1.8085	1.6821	0.3255	0.1032	1.0156	0.9446	0.1828
2	0.9688	0.9532	0.1478	1.9303	0.4129	0.4062	0.0630	0.8226
3	1.9473	0.7664	1.1138	0.1724	0.8258	0.3250	0.4723	0.0731

In Figure 3.5 a plot is shown of the optimal centroid configuration both in its raw and in its uniquely weighted form. The horizontal axis represents the first dimension, while the vertical axis is the third dimension. Comparing these plots with the mould in



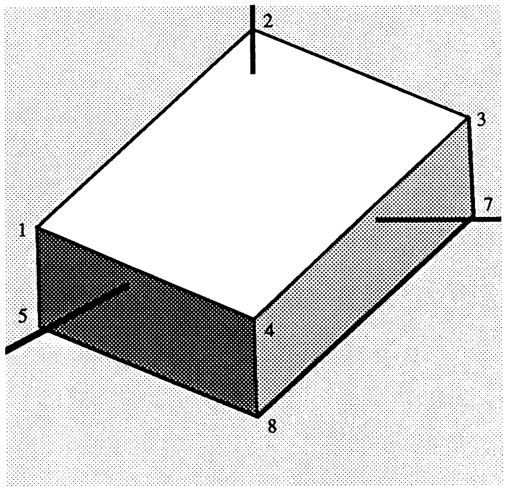


Figure 3.5 Plot of raw centroid configuration **Y** (left), and of uniquely weighted centroid configuration $\mathbf{Y}^*$ (right) from Table 3.6.

Figure 3.4, it becomes clear that we are now dealing with a *permuted* version of the original cube. Specifically, the first dimension in the original in Figure 3.4 has become the third dimension in Figure 3.5, the second dimension has become the first dimension, and the original third dimension now emerges as the second dimension.

Comparing the two plots in Figure 3.5, one notices that the uniquely weighted $\mathbf{Y}^*$ is a blown-up version of the raw **Y**. As can be seen at the bottom of Table 3.6 the raw dimension weights $\mathbf{W}_j$ are accordingly being scaled down to $\mathbf{W}_j^*$ for each j.

Moreover, the coordinates of the stimulus points in $\mathbf{Y}^*$ form the vertices of a cube, as is easily verified by calculating the distances between the stimulus points in Figure 3.5. On the other hand, in the raw centroid configuration **Y**, the stimulus points are the vertices of a parallelepiped, of which all faces are parallelograms, and of which the edges are therefore not orthogonal.

As a logical consequence of the fact that $\mathbf{Y}^*$ is a cube, for each configuration j the ratio of the normalized dimension weights in Table 3.6 is equal to the ratio of the original weights in Table 3.4. This does not hold for the raw dimension weights in Table 3.6. Moreover, if the unique weights $\mathbf{W}_j^*$ in the table are divided by $\sqrt{\text{tr}\,\tilde{\mathbf{X}}'_j\tilde{\mathbf{X}}_j}$ for each j, the sum of the resulting squared weights for each j is equal to one. This follows from the fact that, given n complete configurations, the sum of squared dimension weights $\mathbf{W}_j^*$ corrected for the size of configuration $\tilde{\mathbf{X}}_j$ is equal to the fit of that configuration in the DIMFREE solution (cf. section 3.5.2). Since we have a perfect solution in this example, the fit is equal to one for all four configurations.

In the second part of this section we show the results of an analysis of configurations containing missing data with the DIMFREE model. The four configurations given in Table 3.5 were analyzed treating the following rows as

Table 3.8 History of iterations of DIMFREE algorithm for incomplete configurations in Table 3.5.

	TOTAL LOSS			
Iteration number	new $\mathbf{W}_j$	new $\mathbf{Q}_j$	new $\mathbf{W}_j$	new $\mathbf{Y}$
1	0.33204141	0.29020618	0.28530994	0.17039250
2	0.15547808	0.11105605	0.10885688	0.07072065
3	0.06736023	0.04530308	0.04461105	0.02856942
4	0.02698703	0.01854676	0.01831891	0.01211514
5	0.01135827	0.00818357	0.00808814	0.00562069
6	0.00527978	0.00397960	0.00393580	0.00287635
7	0.00272453	0.00213260	0.00211047	0.00161267
8	0.00154467	0.00124385	0.00123077	0.00097487
9	0.00094396	0.00077505	0.00076603	0.00062339
10	0.00060895	0.00050636	0.00049955	0.00041453
11	0.00040747	0.00034160	0.00033630	0.00028293
12	0.00027926	0.00023537	0.00023125	0.00019645
13	0.00019441	0.00016446	0.00016128	0.00013798
14	0.00013675	0.00011600	0.00011357	0.00009766
15	0.00009689	0.00008235	0.00008052	0.00006951
16	0.00006899	0.00005874	0.00005737	0.00004967
17	0.00004932	0.00004205	0.00004102	0.00003561
18	0.00003536	0.00003018	0.00002942	0.00002559
19	0.00002541	0.00002171	0.00002115	0.00001842
20	0.00001829	0.00001564	0.00001523	0.00001328
21	0.00001319	0.00001129	0.00001098	0.00000959
22	0.00000952	0.00000815	0.00000793	0.00000693
23	0.00000688	0.00000590	0.00000574	0.00000502
24	0.00000498	0.00000427	0.00000415	0.00000363
25	0.00000361	0.00000309	0.00000301	0.00000263
26	0.00000261	0.00000224	0.00000218	0.00000191
27	0.00000189	0.00000162	0.00000158	0.00000138
28	0.00000137	0.00000118	0.00000115	0.00000100
29	0.00000100	0.00000086	0.00000083	0.00000073
30	0.00000072	0.00000062	0.00000060	0.00000053
31	0.00000053	0.00000045	0.00000044	0.00000038
32	0.00000038	0.00000033	0.00000032	0.00000028
33	0.00000028	0.00000024	0.00000023	0.00000020

Table 3.9 Results of DIMFREE analysis of data set in Table 3.5 containing missing data.

$$\mathbf{Y} = \begin{bmatrix} 0.5464 & 0.1470 & -0.0717 \\ -0.0755 & 0.3550 & -0.0294 \\ -0.1821 & 0.1470 & 0.2127 \\ 0.4398 & -0.0609 & 0.1711 \\ 0.1820 & -0.1470 & -0.2128 \\ -0.4398 & 0.0609 & -0.1704 \\ -0.5463 & -0.1470 & 0.0709 \\ 0.0755 & -0.3550 & 0.0296 \end{bmatrix} \qquad \mathbf{Y}^* = \begin{bmatrix} 0.5303 & 0.2500 & -0.1787 \\ -0.0732 & 0.6036 & -0.0733 \\ -0.1767 & 0.2500 & 0.5306 \\ 0.4269 & -0.1035 & 0.4267 \\ 0.1766 & -0.2500 & -0.5308 \\ -0.4269 & 0.1035 & -0.4250 \\ -0.5303 & -0.2500 & 0.1768 \\ 0.0733 & -0.6035 & 0.0738 \end{bmatrix}$$

$$Q_1 = \begin{bmatrix} 0.8779 & -0.3944 & 0.2717 \\ 0.3907 & 0.9178 & 0.0698 \\ -0.2769 & 0.0449 & 0.9599 \end{bmatrix} \qquad Q_2 = \begin{bmatrix} 0.7129 & -0.4162 & 0.5643 \\ 0.5037 & 0.8639 & 0.0008 \\ -0.4879 & 0.2837 & 0.8255 \end{bmatrix}$$

$$Q_3 = \begin{bmatrix} 0.9602 & -0.2772 & -0.0342 \\ 0.2728 & 0.9570 & -0.0987 \\ 0.0601 & 0.0854 & 0.9945 \end{bmatrix} \qquad Q_4 = \begin{bmatrix} 0.9504 & -0.2163 & 0.2237 \\ 0.2583 & 0.9493 & -0.1795 \\ -0.1735 & 0.2283 & 0.9580 \end{bmatrix}$$

	raw dimension weights $\mathbf{w}_j$				unique dimension weights $\mathbf{w}^*_j$			
	configuration				configuration			
dimension	1	2	3	4	1	2	3	4
1	0.4827	0.3361	1.3530	1.3169	0.4974	0.3463	1.3940	1.3568
2	1.6913	0.4710	1.1843	0.2050	0.9947	0.2770	0.6965	0.1206
3	0.3101	2.1582	0.1757	0.7507	0.1243	0.8653	0.0704	0.3010

missing: rows 1, 3, 6 and 8 of configuration 2, rows 1, 3, 5 and 7 of configuration 3, and rows 3 and 7 of configuration 4.

A generalized Procrustes analysis of these four configurations containing missing data yielded a loss of 0.8280, meaning that the GPA model now accounts for (100 x (4 – 0.8280)/4) = 79% of the total sum of squares of the four configurations.

Submitting the optimally rotated and rescaled configurations from GPA to the algorithm for the DIMFREE model, we again obtained the expected results. The algorithm converged in 33 iterations, yielding a perfect solution. The history of iterations for this data set is shown in Table 3.8, and the optimal transformation parameters of the DIMFREE solution are given in Table 3.9.

Within rounding errors, and up to permutations of the three dimensions, the uniquely weighted centroid configuration $\mathbf{Y}^*$ in Table 3.9 is equal to $\mathbf{Y}^*$ in Table 3.6. This shows that the DIMFREE algorithm also recovers the original mould **AT** of this artificial data set if missing data are introduced.

Due to the fact that we are dealing with incomplete configurations in this example, the sum of the squared dimension weights $\mathbf{W}_j^*$ shown in Table 3.9 is no longer related to the fit of each configuration $\tilde{\mathbf{X}}_j$ in the DIMFREE solution.

3.6.2 DIMIDIO analysis of a constructed data set

The data discussed in this section were again constructed such that a perfect solution must exist when the admissible transformations are those corresponding to dimension weighting model (3.5). We used the mould and transformation parameters given in Table 3.4 to construct four configurations by calculating $\mathbf{X}_j = s_j(\mathbf{AR}_j\mathbf{W}_j - \mathbf{1u}_j')$ for j = 1, ..., 4. Thus, the coordinates of the vertices of the cube **A** were first idiosyncratically rotated by $\mathbf{R}_j$, then differentially weighted by $\mathbf{W}_j$, and finally idiosyncratically translated and centrally dilated. This resulted in the data given in Table 3.10.

A generalized Procrustes analysis of these four complete configurations yielded a solution in six iterations, with a loss of 0.8335. This implies that the GPA model accounts for 79% of the total variation in the configurations in Table 3.10.

Submitting the results from GPA to the DIMFREE algorithm of section 3.3.8, it was found that this algorithm converged in 58 iterations, giving a loss of 0.1784. Thus, the DIMFREE model yielded a solution which accounts for 96% of the total variation, 17% more than the GPA model.

Table 3.10 Constructed data for analysis with DIMIDIO model.

$$\mathbf{X}_1 = \begin{bmatrix} -12.9338 & -0.8184 & -5.1471 \\ -1.3868 & -0.4520 & -4.4265 \\ -2.2232 & -0.7601 & 1.4307 \\ -13.7702 & -1.1264 & 0.7101 \\ -9.7768 & -2.2399 & -6.2307 \\ 1.7702 & -1.8736 & -5.5101 \\ 0.3436 & -2.1816 & 0.3471 \\ -10.6132 & -2.5480 & -0.3735 \end{bmatrix} \quad \mathbf{X}_2 = \begin{bmatrix} 0.6774 & 0.4417 & 0.8746 \\ 0.9783 & 0.1422 & 1.1815 \\ 1.2218 & 0.9034 & 0.9273 \\ 0.9209 & 1.2029 & 0.6204 \\ 0.7782 & -0.5034 & 0.5727 \\ 1.0791 & -0.8029 & 0.8796 \\ 1.3226 & -0.0417 & 0.6254 \\ 1.0217 & 0.2578 & 0.3185 \end{bmatrix}$$

$$\mathbf{X}_3 = \begin{bmatrix} -0.1564 & -2.9912 & -2.9643 \\ -0.5420 & 0.9484 & -2.9883 \\ 2.0971 & 1.5866 & -2.7842 \\ 2.4826 & -2.3530 & -2.7602 \\ -2.4971 & 0.4134 & -3.2158 \\ -2.8826 & 4.3530 & -3.2398 \\ -0.2436 & 4.9912 & -3.0357 \\ 0.1420 & 1.0516 & -3.0117 \end{bmatrix} \quad \mathbf{X}_4 = \begin{bmatrix} -3.1684 & -0.8547 & -3.3878 \\ -3.1241 & 1.8452 & 3.6610 \\ -3.0944 & 3.7627 & 4.3163 \\ -3.1387 & 1.0628 & -2.7325 \\ -3.3056 & -0.5627 & -4.3163 \\ -3.2613 & 2.1372 & 2.7325 \\ -3.2316 & 4.0547 & 3.3878 \\ -3.2759 & 1.3548 & -3.6610 \end{bmatrix}$$

Table 3.11 Results of DIMIDIO analysis of complete data set in Table 3.10.

$$\mathbf{Y} = \begin{bmatrix} -0.3252 & -0.1183 & -0.0534 \\ 0.1595 & -0.0335 & -0.0883 \\ 0.1088 & 0.2432 & -0.0368 \\ -0.3760 & 0.1584 & -0.0019 \\ -0.1088 & -0.2432 & 0.0368 \\ 0.3760 & -0.1584 & 0.0019 \\ 0.3252 & 0.1183 & 0.0534 \\ -0.1595 & 0.0335 & 0.0883 \end{bmatrix} \qquad \mathbf{Y}^* = \begin{bmatrix} -0.4312 & -0.2653 & -0.3445 \\ 0.2115 & -0.0751 & -0.5697 \\ 0.1442 & 0.5456 & -0.2378 \\ -0.4985 & 0.3554 & -0.0126 \\ -0.1442 & -0.5456 & 0.2378 \\ 0.4985 & -0.3554 & 0.0126 \\ 0.4312 & 0.2653 & 0.3445 \\ -0.2115 & 0.0751 & 0.5697 \end{bmatrix}$$

$$\mathbf{S}_1 = \begin{bmatrix} 0.9881 & -0.0571 & 0.1431 \\ 0.0933 & 0.9608 & -0.2610 \\ -0.1226 & 0.2713 & 0.9547 \end{bmatrix} \qquad \mathbf{S}_2 = \begin{bmatrix} -0.5827 & -0.3612 & 0.7280 \\ 0.7698 & 0.0418 & 0.6369 \\ -0.2605 & 0.9315 & 0.2537 \end{bmatrix}$$

$$\mathbf{S}_3 = \begin{bmatrix} 0.9208 & 0.1667 & -0.3527 \\ -0.1797 & 0.9837 & -0.0043 \\ 0.3462 & 0.0674 & 0.9357 \end{bmatrix} \qquad \mathbf{S}_4 = \begin{bmatrix} 0.8348 & -0.0475 & 0.5485 \\ 0.4548 & 0.6210 & -0.6384 \\ -0.3103 & 0.7824 & 0.5400 \end{bmatrix}$$

$$\mathbf{Q}_1 = \begin{bmatrix} 0.9969 & -0.0615 & 0.0493 \\ 0.0683 & 0.9864 & -0.1496 \\ -0.0394 & 0.1525 & 0.9875 \end{bmatrix} \qquad \mathbf{Q}_2 = \begin{bmatrix} -0.6873 & -0.3116 & 0.6561 \\ 0.7074 & -0.0819 & 0.7021 \\ -0.1651 & 0.9467 & 0.2767 \end{bmatrix}$$

$$\mathbf{Q}_3 = \begin{bmatrix} 0.9844 & 0.1567 & -0.0803 \\ -0.1574 & 0.9875 & -0.0022 \\ 0.0790 & 0.0148 & 0.9968 \end{bmatrix} \qquad \mathbf{Q}_4 = \begin{bmatrix} 0.9506 & -0.2234 & 0.2157 \\ 0.2986 & 0.8482 & -0.4376 \\ -0.0852 & 0.4803 & 0.8729 \end{bmatrix}$$

	dimension weights $\mathbf{W}_j$					dimension weights $\mathbf{W}^*_j = \mathbf{W}_j/(\sqrt{\text{tr}\, \tilde{\mathbf{X}}_j'\tilde{\mathbf{X}}_j})$			
	configuration					configuration			
dimension	1	2	3	4		1	2	3	4
1	0.9815	0.8226	0.9226	0.8844		0.8889	0.8900	0.8766	0.9775
2	0.4908	0.3290	0.5065	0.1903		0.4444	0.3560	0.4812	0.2103
3	0.1227	0.2632	0.0079	0.0149		0.1111	0.2848	0.0075	0.0165
					Σw^{*2}_{jk}	1.0000	1.0000	1.0000	1.0000

Thereafter, submitting the optimally rotated and rescaled configurations $s_j\mathbf{X}_j\mathbf{R}_j\mathbf{K}$ and the optimal centroid configuration **ZK** of the GPA solution to the DIMIDIO algorithm of section 3.4.7, this algorithm immediately reached a perfect solution in the first iteration. The results of this analysis are given in Table 3.11. As expected, the DIMIDIO model accounts for all the variation in this constructed data set. Of course, in practice one would choose for the results of the DIMFREE model, since this model is much more parsimonious and accounts for almost the same percentage of total variation as the DIMIDIO model. But our main purpose here is to investigate whether the DIMIDIO algorithm finds a perfect solution when it exists.

At the bottom of Table 3.11 is shown that, since all configurations are complete in this example, the sum of squared dimension weights corrected for the size of configuration $\tilde{\mathbf{X}}_j$ is equal to the fit of each configuration. Because we have a perfect solution, the fit is equal to one for all four configurations.

Introducing the same missing data in the configurations of Table 3.10 as in section 3.6.1, that is, treating stimuli 1, 3, 6 and 8 of configuration 2, stimuli 1, 3, 5 and 7 of configuration 3, and stimuli 3 and 7 of configuration 4 as missing, the GPA model yielded a loss of 0.6770. Thus, this model accounts for 83% of the variation in these incomplete configurations.

Submitting the optimally rotated and rescaled configurations $s_j\mathbf{X}_j\mathbf{R}_j\mathbf{K}$ and **ZK** from GPA to the DIMFREE algorithm, it was found that this algorithm took 108 iterations to converge. At convergence the loss was 0.0773, meaning that the DIMFREE model accounts for 98% of the total variation.

The DIMIDIO algorithm of section 3.4.7 again converged to a perfect solution, in 48 iterations. The history of iterations for this data set is shown in Table 3.12, and the results of the analysis are given in Table 3.13. In Figure 3.6 we have plotted the coordinates of the optimal centroid configuration $\mathbf{Y}^*$ in Table 3.11 (i.e., after DIMIDIO analysis of the four complete configurations) together with the coordinates of $\mathbf{Y}^*$ in Table 3.13 (i.e., after DIMIDIO analysis of the four incomplete configurations). For the example discussed in the previous section about DIMFREE we saw that the optimal uniquely weighted centroid configurations $\mathbf{Y}^*$ for complete and incomplete data were identical up to permutations and reflections of the columns of the two $\mathbf{Y}^*$'s. In DIMIDIO the centroid configuration $\mathbf{Y}^*$ given in Table 3.13 is a

Table 3.12 History of iterations of DIMIDIO analysis of incomplete data set in Table 3.10.

Iteration number	TOTAL LOSS new $\mathbf{B}_j$	TOTAL LOSS new **Y**	Iteration number	TOTAL LOSS new $\mathbf{B}_j$	TOTAL LOSS new **Y**
1	0.04255481	0.01220214	25	0.00004515	0.00004023
2	0.00746972	0.00580469	26	0.00003586	0.00003204
3	0.00494796	0.00439940	27	0.00002864	0.00002571
4	0.00400550	0.00368420	28	0.00002310	0.00002090
5	0.00340545	0.00314840	29	0.00001897	0.00001670
6	0.00290998	0.00268247	30	0.00001495	0.00001319
7	0.00246822	0.00226339	31	0.00001167	0.00001049
8	0.00207090	0.00188833	32	0.00000952	0.00000866
9	0.00171806	0.00155823	33	0.00000790	0.00000721
10	0.00141051	0.00127326	34	0.00000659	0.00000602
11	0.00114752	0.00103174	35	0.00000551	0.00000504
12	0.00092649	0.00083032	36	0.00000461	0.00000421
13	0.00074344	0.00066458	37	0.00000385	0.00000352
14	0.00059369	0.00052967	38	0.00000322	0.00000295
15	0.00047236	0.00042080	39	0.00000270	0.00000247
16	0.00037479	0.00033352	40	0.00000226	0.00000206
17	0.00029678	0.00026390	41	0.00000189	0.00000173
18	0.00023468	0.00020858	42	0.00000158	0.00000145
19	0.00018540	0.00016474	43	0.00000132	0.00000121
20	0.00014640	0.00013006	44	0.00000111	0.00000101
21	0.00011557	0.00010269	45	0.00000093	0.00000085
22	0.00009125	0.00008110	46	0.00000078	0.00000071
23	0.00007209	0.00006409	47	0.00000065	0.00000059
24	0.00005700	0.00005072	48	0.00000054	0.00000050

Table 3.13 Results of DIMIDIO analysis of incomplete data set in Table 3.10.

$$\mathbf{Y} = \begin{bmatrix} -0.4103 & -0.1861 & -0.0702 \\ 0.2804 & -0.0124 & -0.0940 \\ 0.2190 & 0.3092 & -0.0545 \\ -0.4716 & 0.1361 & -0.0313 \\ -0.2190 & -0.3094 & 0.0547 \\ 0.4716 & -0.1357 & 0.0308 \\ 0.4102 & 0.1854 & 0.0710 \\ -0.2804 & 0.0129 & 0.0935 \end{bmatrix} \qquad \mathbf{Y}^* = \begin{bmatrix} -0.4256 & -0.1890 & -0.3963 \\ 0.2635 & -0.1166 & -0.5412 \\ 0.2662 & 0.5092 & -0.2117 \\ -0.4227 & 0.4379 & -0.0702 \\ -0.2662 & -0.5096 & 0.2126 \\ 0.4228 & -0.4370 & 0.0676 \\ 0.4254 & 0.1876 & 0.4006 \\ -0.2634 & 0.1175 & 0.5386 \end{bmatrix}$$

$$\mathbf{S}_1 = \begin{bmatrix} 0.9965 & 0.0802 & -0.0236 \\ -0.0836 & 0.9584 & -0.2728 \\ 0.0008 & 0.2738 & 0.9618 \end{bmatrix} \qquad \mathbf{S}_2 = \begin{bmatrix} -0.4009 & -0.4644 & 0.7897 \\ 0.8590 & 0.1091 & 0.5003 \\ -0.3185 & 0.8789 & 0.3552 \end{bmatrix}$$

$$\mathbf{S}_3 = \begin{bmatrix} 0.4255 & -0.8985 & 0.1074 \\ -0.8347 & -0.3439 & 0.4301 \\ 0.3495 & 0.2727 & 0.8964 \end{bmatrix} \qquad \mathbf{S}_4 = \begin{bmatrix} 0.9336 & -0.0321 & 0.3569 \\ 0.2968 & 0.6274 & -0.7199 \\ -0.2008 & 0.7780 & 0.5953 \end{bmatrix}$$

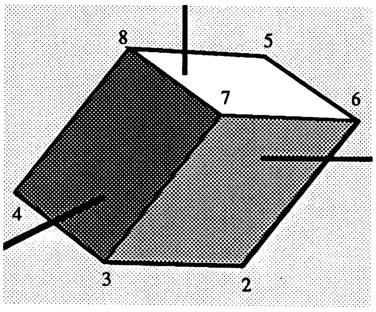
$$\mathbf{Q}_1 = \begin{bmatrix} 0.9864 & -0.1639 & -0.0133 \\ 0.1600 & 0.9754 & -0.1518 \\ 0.0379 & 0.1476 & 0.9883 \end{bmatrix} \qquad \mathbf{Q}_2 = \begin{bmatrix} -0.6341 & -0.5391 & 0.5543 \\ 0.6989 & -0.0930 & 0.7091 \\ -0.3307 & 0.8371 & 0.4357 \end{bmatrix}$$

$$\mathbf{Q}_3 = \begin{bmatrix} 0.9643 & -0.1556 & 0.2144 \\ -0.2455 & -0.2210 & 0.9439 \\ 0.0994 & 0.9628 & 0.2513 \end{bmatrix} \qquad \mathbf{Q}_4 = \begin{bmatrix} 0.9699 & -0.1969 & 0.1433 \\ 0.2240 & 0.9523 & -0.2074 \\ -0.0956 & 0.2333 & 0.9677 \end{bmatrix}$$

dimension weights

$\mathbf{W}_j$

	configuration			
dimension	1	2	3	4
1	1.1701	0.8825	1.0959	1.2946
2	0.5851	0.3528	0.1268	0.2795
3	0.1463	0.2827	0.0000	0.0219



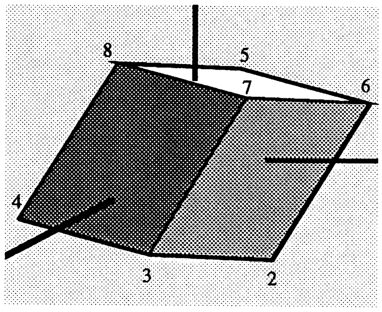


Figure 3.6 Optimal $\mathbf{Y}^*$ after DIMIDIO analysis of complete data set (left), and of incomplete data set (right).

rotated version of the $\mathbf{Y}^*$ given in Table 3.11. This is an immediate consequence of the fact that $\mathbf{Y}^*$ in DIMIDIO is only determined up to a rotation (see section 3.4.6).

We finally note that the sum of squared dimension weights in Table 3.13 is no longer related to the fit of the configurations in the DIMIDIO solution, due to the fact that we are now dealing with incomplete configurations.